



Recommending Items to Users: An Explore Exploit Perspective

Deepak Agarwal, Director Machine Learning and
Relevance Science, LinkedIn, USA

CIKM, 2013

Disclaimer

- Opinions expressed are mine and in no way represent the official position of LinkedIn
- Material inspired by work done at LinkedIn and Yahoo!

Main Collaborators: several others at both Y! and LinkedIn

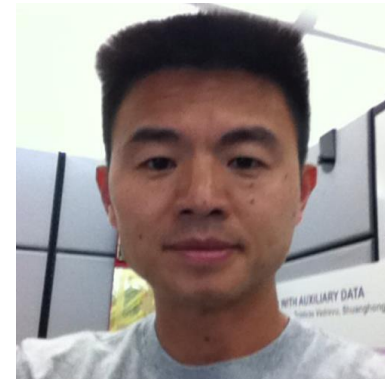
- I won't be here without them, extremely lucky to work with such talented individuals



Bee-Chung Chen



Liang Zhang



Bo Long



Jonathan Traupman

Item Recommendation problem
Arises in both advertising and content


*Serve the “best” items (in different contexts) to users in an **automated** fashion to optimize long-term business objectives*


Business Objectives

User engagement, Revenue,...


LinkedIn Today: Content Module

All Updates ▾


 LinkedIn Today recommends this news for you



Please Feed The Meters: The Next Parking Revolution
collectorsweekly.com




How B-Schools Pick Their Students
John A. Byrne



Do you hoard your new underwear?
Gretchen Rubin

More Influencer Posts ►



Objective: Serve content to maximize engagement metrics like CTR (or weighted CTR)

Similar problem: Content recommendation on Yahoo! front page

The image shows a screenshot of the Yahoo! front page from July 4, 2011. Several areas are highlighted with colored boxes and labels to illustrate content recommendation slots:

- Today module:** A red box highlights the main featured story, "World Cup octopus could make millions," which includes a large image of Paul the octopus and a sub-headline about its demand. Below this story are four smaller image thumbnails labeled **F1**, **F2**, **F3**, and **F4**.
- Trending Now:** A yellow box highlights the "TRENDING NOW" list on the right side of the page, which includes items like "Kourtney Kardash...", "Anna Chapman", "Al Pacino", "French Toast Rec...", "Nina Garcia", "Susan Boyle", "Job Search", "Yogi Berra", "Philippines Typh...", and "Sunscreen".
- NEWS:** A green box highlights the "NEWS" section at the bottom of the page, which lists various headlines such as "9 killed, 10 missing as typhoon lashes Philippines" and "Testing delayed on tighter cap for Gulf oil well".

Annotations and text on the right side of the image provide context for these slots:

- An arrow points from the text "Recommend content links (out of 30-40, editorially programmed)" to the "More on the octopus" link in the Today module.
- The text "4 slots exposed, F1 has maximum exposure" is positioned near the Today module.
- The text "Routes traffic to other Y! properties" is positioned near the NEWS section.

LinkedIn Ads: Match ads to users visiting LinkedIn

The screenshot displays a web browser window with the LinkedIn homepage. The browser's address bar shows 'www.linkedin.com' and the page title is '(22) Welcome, Deepak! | LinkedIn'. The navigation bar includes links for Home, Profile, Contacts, Groups, Jobs, Inbox (50), Companies, News, and More. A search bar is located on the right side of the navigation bar. The main content area features a list of updates from users who are now connected to specific individuals. The updates include:

- Jerry Ye** is now connected to **Nam Nguyen**, (Feeling Like) an Intern @Facebook. Send a message • 5 hours ago.
- Indrajit Chatterjee** is now connected to **Anamitra Chatterjee**, Executive & Career Coach, Leadership Facilitator. Send a message • 5 hours ago.
- Romita Mukherjee** is now connected to **Sudha Narayanan**, Assistant Professor at IGIDR. Send a message • 6 hours ago.

Below the updates is a link to 'SHOW MORE UPDATES'. On the right side, there is a section titled 'ADS BY LINKEDIN MEMBERS' featuring three advertisements:

- Are You A Director?** Apply Now to the Worldwide Who's Who network for Distinguished Individuals.
- KAE** Mobile device strategy. Insight, strategy & innovation specialists for the mobile industry.
- aria** Cloud Based Billing eBook. Learn How to Build a well-rounded Cloud based billing platform for Free.

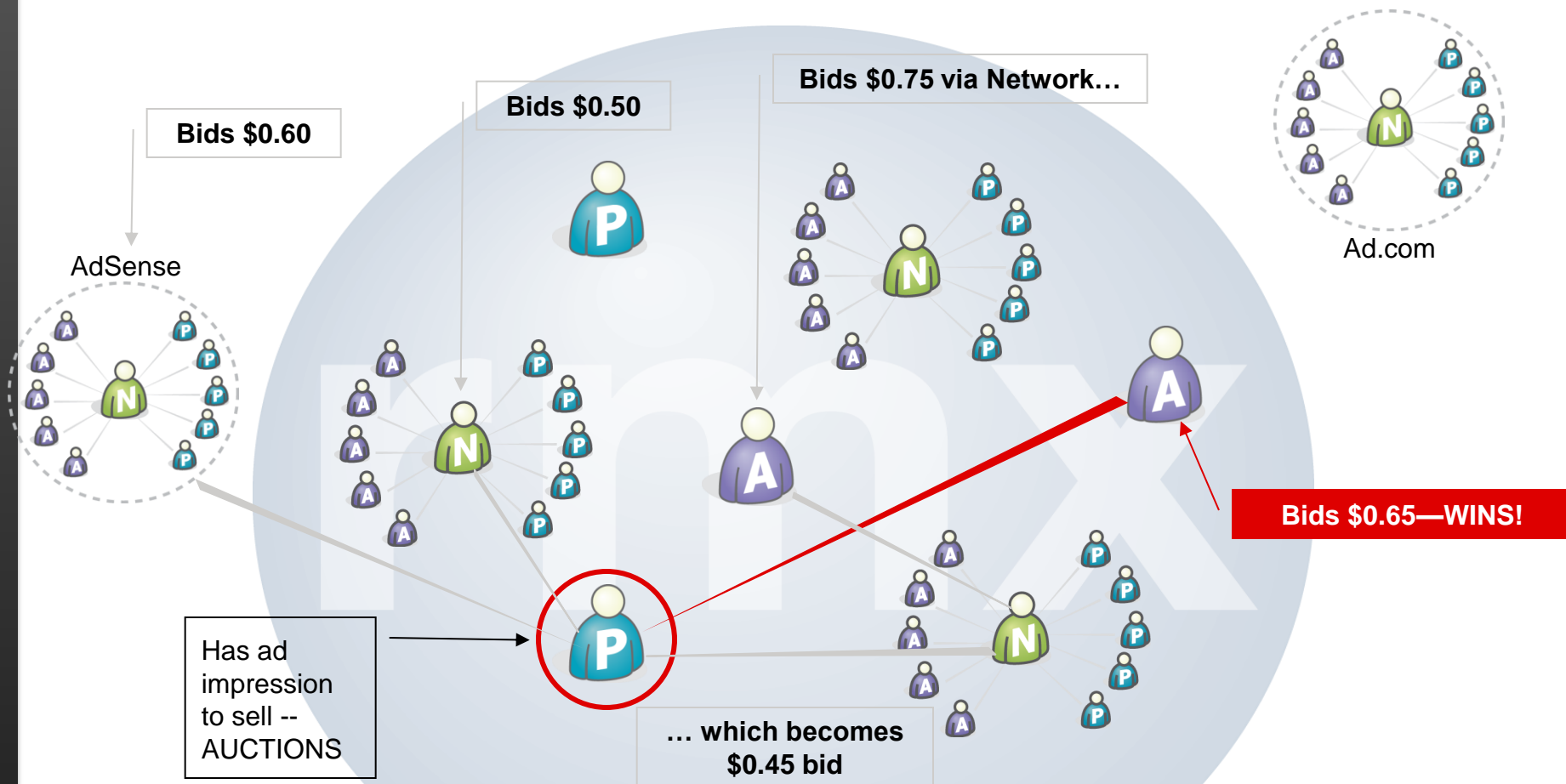
At the bottom of the page, there is a large advertisement for the Hyundai Azera, featuring a silver car and the text 'THE BEAUTIFULLY RESPONSIVE AZERA. CLICK TO EXPAND'. The footer includes links for Help Center, About, Blog, Careers, Advertising, Recruiting Solutions, Tools, Mobile, Developers, Publishers, Language, and Upgrade Your Account. The LinkedIn Corporation © 2012 copyright notice is also present.

Feedback

pre
df
2pr
pof
Let
rwa
y of
.pd
vi.p

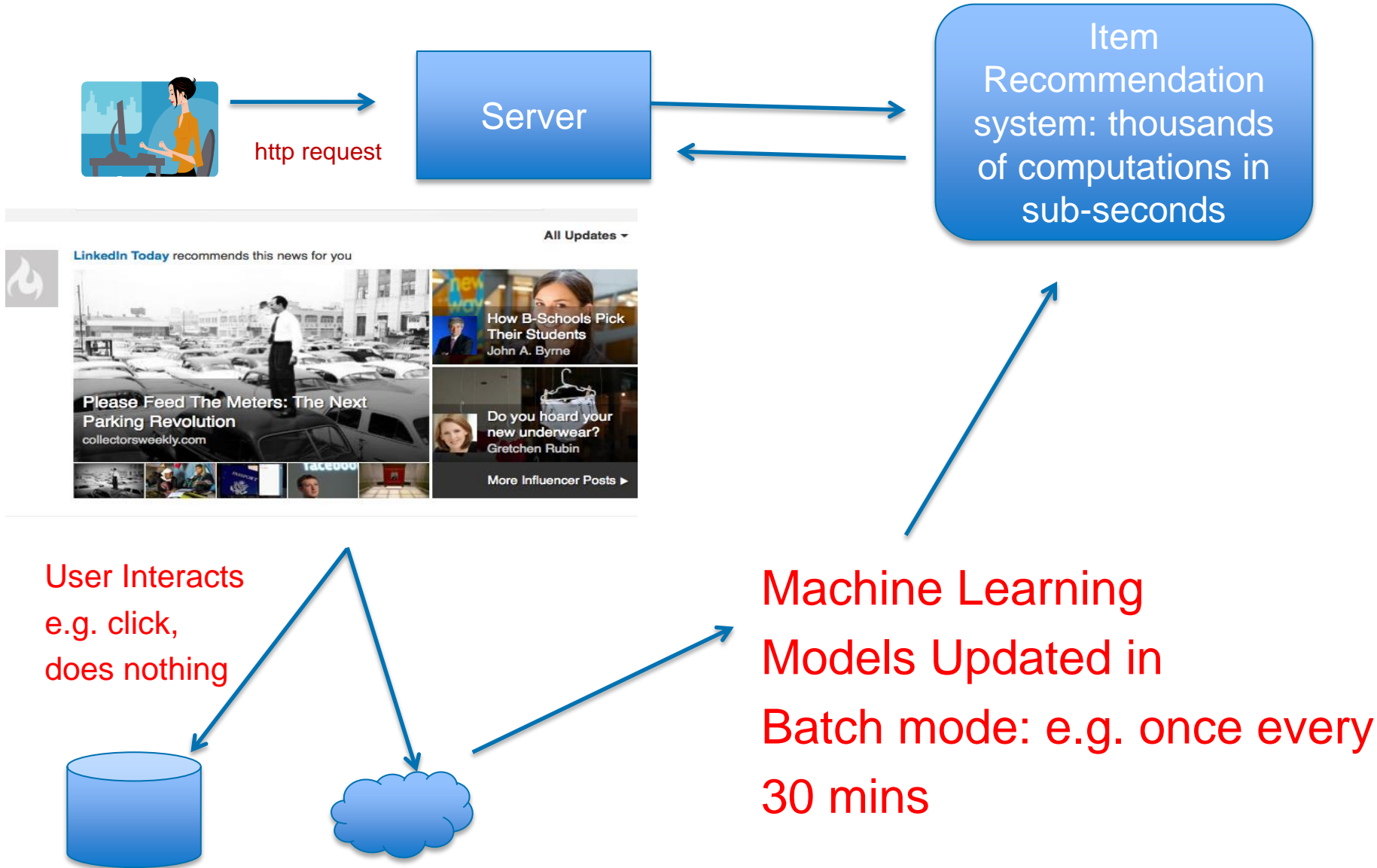


Right Media Ad Exchange: Unified Marketplace

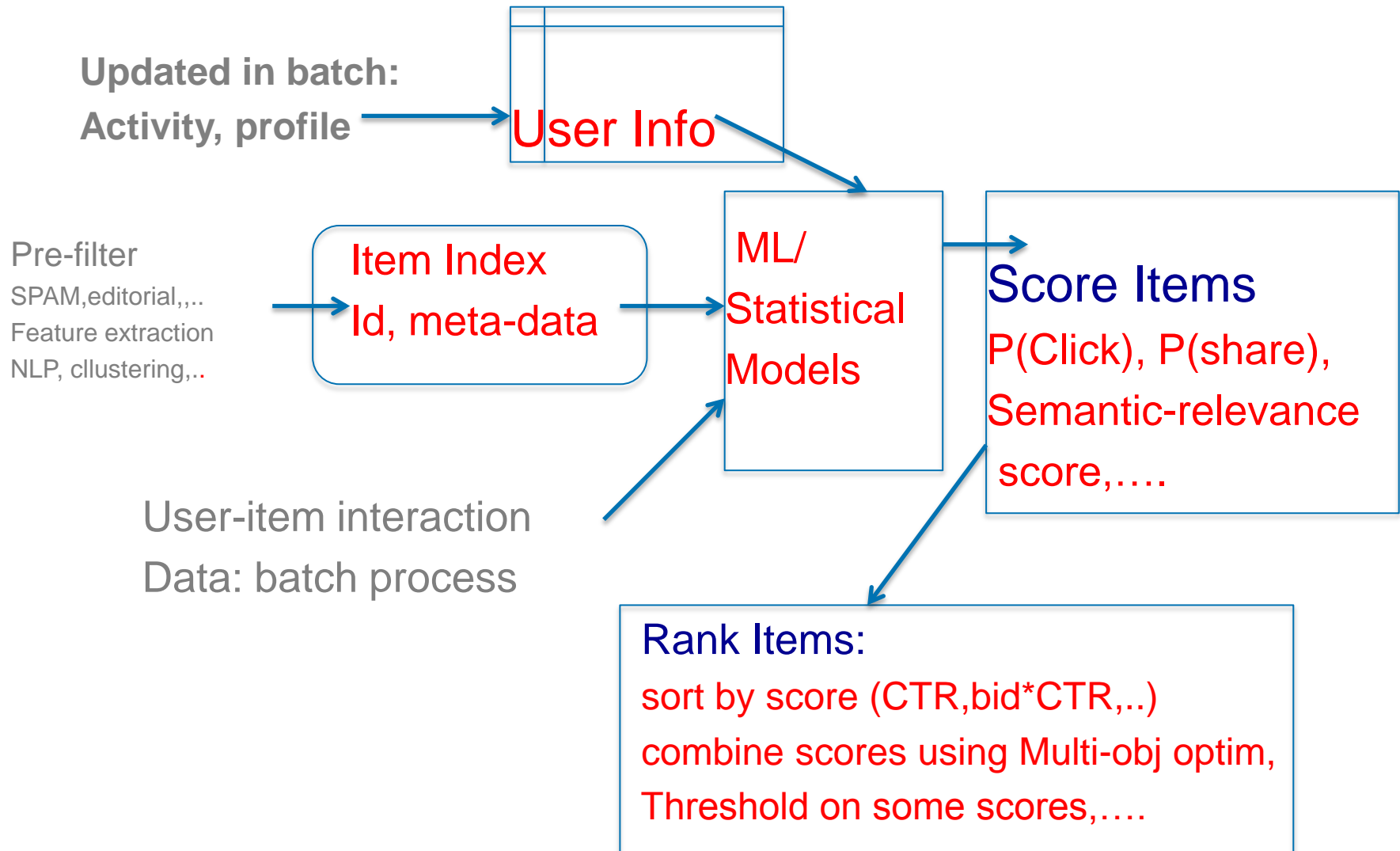


Match ads to page views on publisher sites

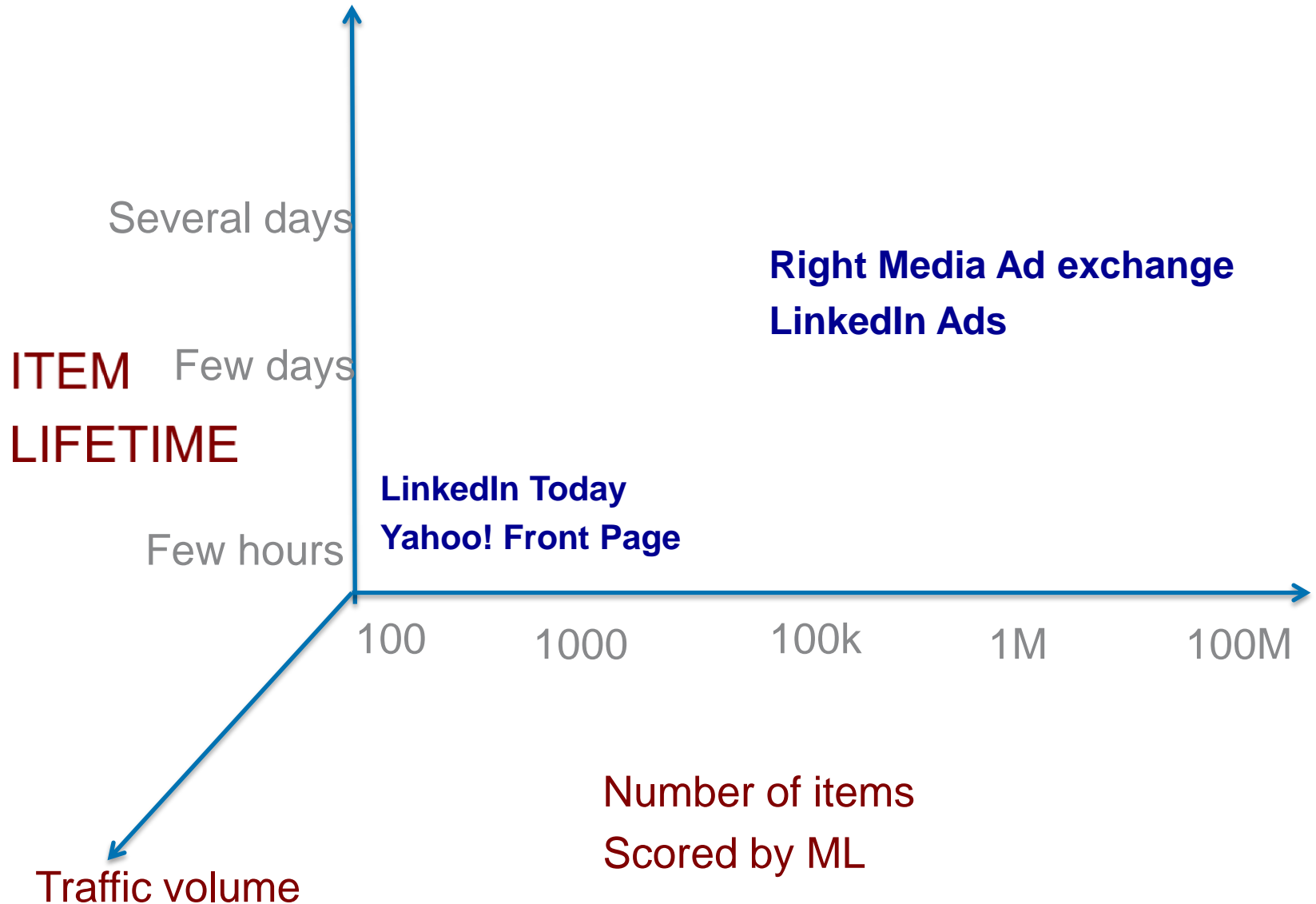
High level picture



High level overview: Item Recommendation System



ML/Statistical models for scoring



Explore/Exploit deployments

- Yahoo! Front page Today Module (2008-2011): 300% improvement in click-through rates
 - Similar algorithms delivered via a self-serve platform, adopted by several Yahoo! Properties (2011): Significant improvement in engagement across Yahoo! Network
- Fully deployed on LinkedIn Today Module (2012): Significant improvement in click-through rates (numbers not revealed due to reasons of confidentiality)
- Yahoo! RightMedia exchange (2012): Fully deployed algorithms to estimate response rates (CTR, conversion rates). Significant improvement in revenue (numbers not revealed due to reasons of confidentiality)
- LinkedIn self-serve ads (2012): Tests on large fraction of traffic shows significant improvements. Fully deployed.

Statistical Problem

- Rank items (from an admissible pool) for user visits in some context to maximize a utility of interest
- Examples of utility functions
 - Click-rates (CTR)
 - Share-rates ($\text{CTR} * [\text{Share}|\text{Click}]$)
 - Revenue per page-view = $\text{CTR} * \text{bid}$ (more complex due to second price auction)
- CTR is a fundamental measure that opens the door to a more principled approach to rank items
- Converge rapidly to maximum utility items
 - Sequential decision making process (explore/exploit)

LinkedIn Today, Yahoo! Today Module:

Choose Items to maximize CTR

This is an “Explore/Exploit” Problem



Algorithm selects item j from a set of candidates



User i visits
with

user features

(e.g., industry,
behavioral features,
Demographic
features,.....)

—————→ (i, j) : response y_{ij}
(click or not)

Which item should we select?

- The item with highest predicted CTR
- An item for which we need data to predict its CTR

Exploit

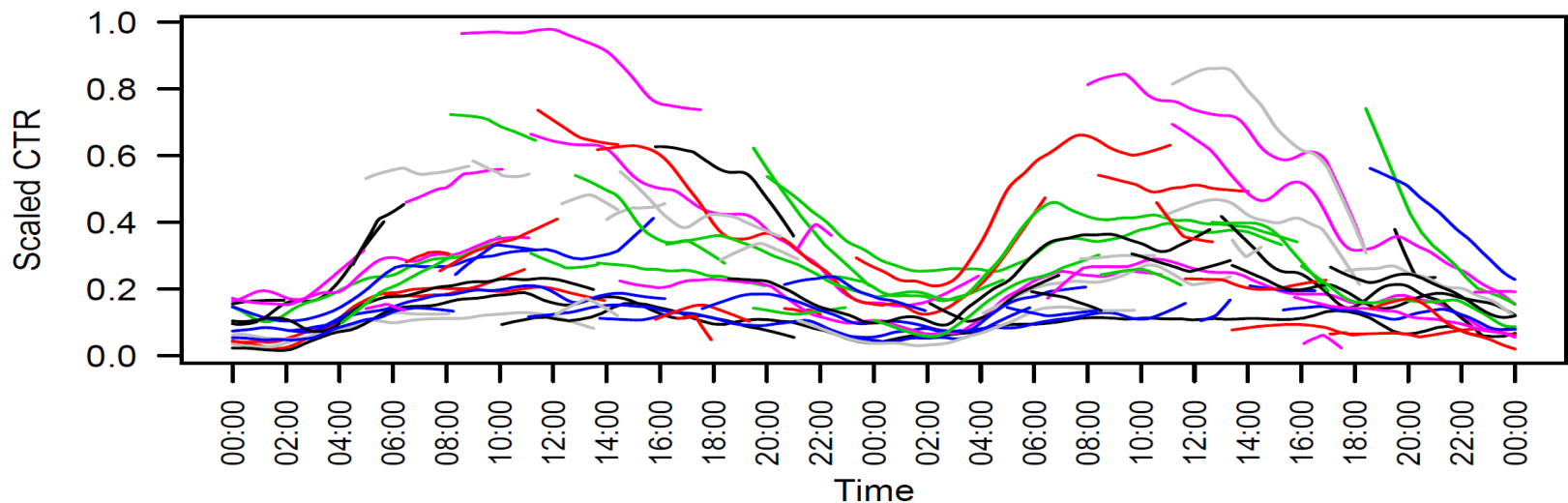
Explore

The Explore/Exploit Problem (to maximize CTR)

- Problem definition: Pick k items from a pool of N for a large number of serves to maximize the number of clicks on the picked items
- Easy!? Pick the items having the highest click-through rates (CTRs)
- But ...
 - The system is highly **dynamic**:
 - Items come and go with short lifetimes
 - CTR of each item may change over time
 - How much traffic should be allocated to **explore** new items to achieve optimal performance ?
 - Too little → Unreliable CTR estimates due to “**starvation**”
 - Too much → Little traffic to **exploit** the high CTR items

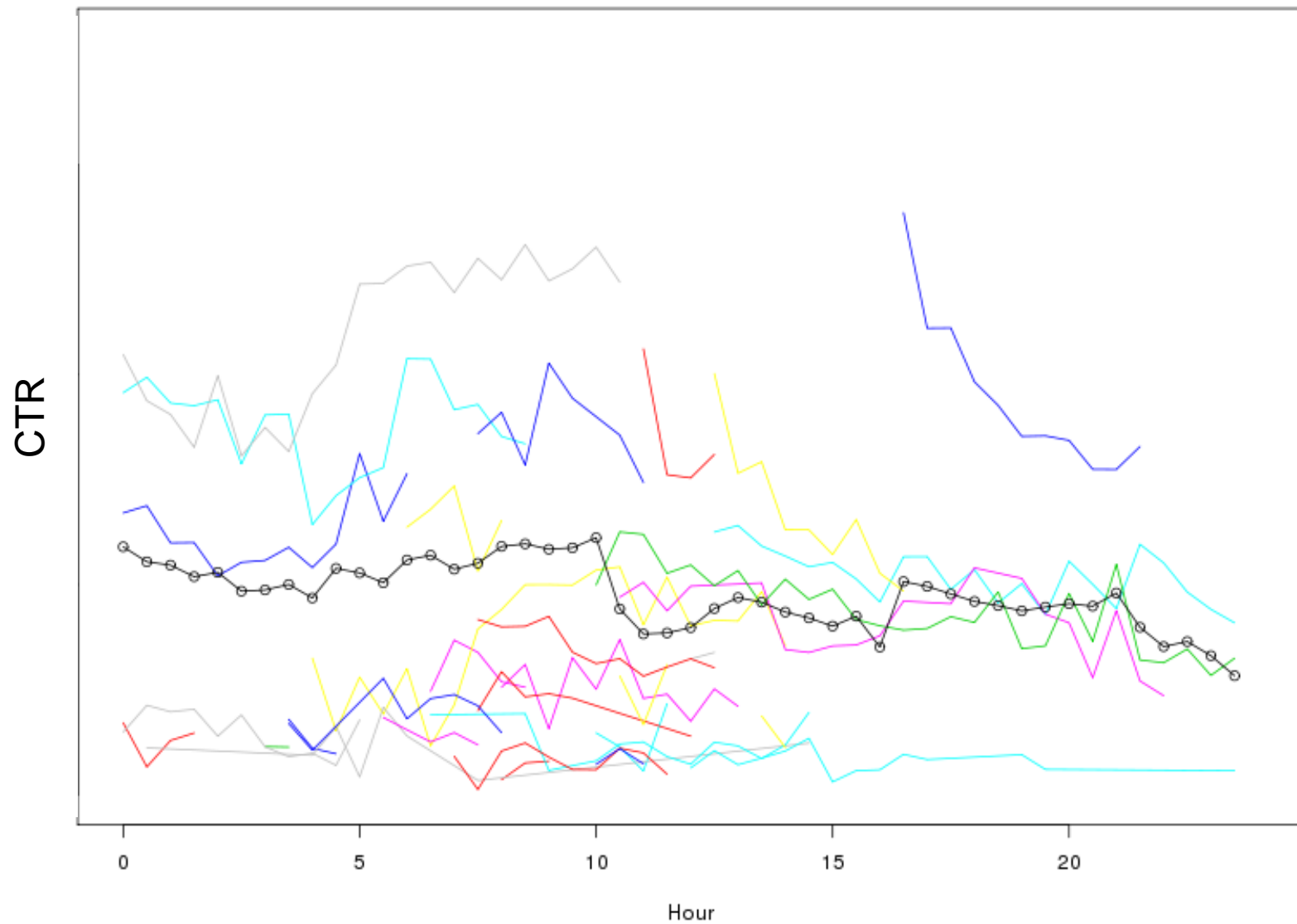
Y! front Page Application

- Simplify: Maximize CTR on first slot (F1)
- Item Pool
 - Editorially selected for high quality and brand image
 - Few articles in the pool but item pool dynamic



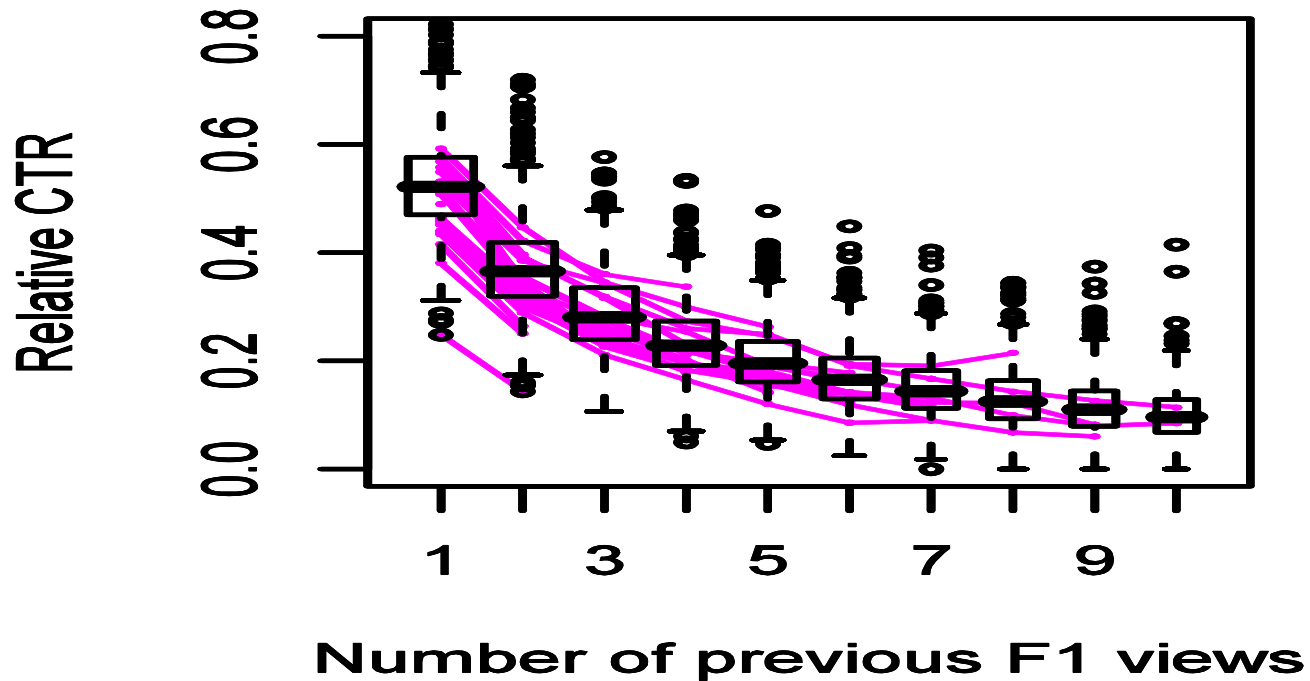
CTR Curves of Items on LinkedIn Today

2012-04-06



Impact of repeat item views on a given user

- Same user is shown an item multiple times (despite not clicking)



Simple algorithm to estimate most popular item with small but dynamic item pool

- Simple Explore/Exploit scheme

- ϵ % explore: with a small probability (e.g. 5%), choose an item at random from the pool
- $(100-\epsilon)$ % exploit: with large probability (e.g. 95%), choose highest scoring CTR item

- Temporal Smoothing

- Item CTRs change over time, provide more weight to recent data in estimating item CTRs
 - Kalman filter, moving average

- Discount item score with repeat views

- $\text{CTR}(\text{item})$ for a given user drops with repeat views by some “discount” factor (estimated from data)

- Segmented most popular

- Perform separate most-popular for each user segment

More economical exploration? Better bandit solutions

- Consider two armed problem



p_1

>

p_2

(unknown payoff probabilities)

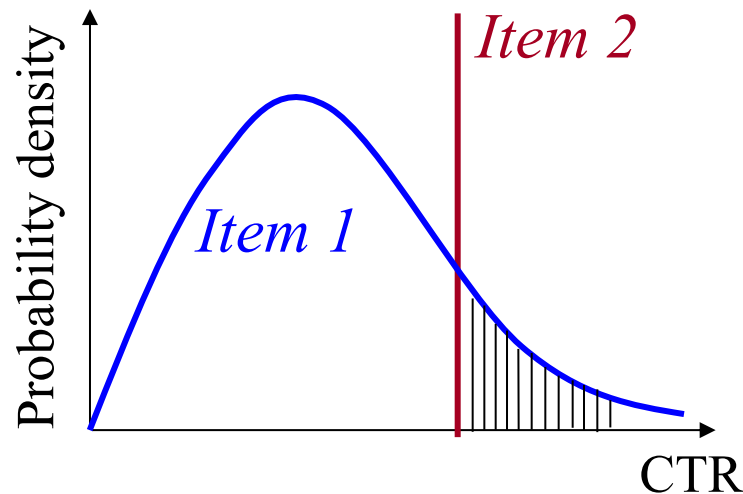
The gambler has 1000 plays, what is the best way to experiment ?
(to maximize total expected reward)

This is called the “multi-armed bandit” problem, have been studied for a long time.

Optimal solution: Play the arm that has maximum *potential of being good*
Optimism in the face of uncertainty

Item Recommendation: Bandits?

- *Two Items*: Item 1 **CTR= 2/100** ; Item 2 **CTR= 250/10000**
 - *Greedy*: Show Item 2 to all; not a good idea
 - Item 1 CTR estimate noisy; item could be potentially better
 - Invest in Item 1 for better overall performance on average



- Exploit what is known to be good, explore what is potentially good

Explore/Exploit with large item pool/personalized recommendation

- Obtaining optimal solution difficult in practice
- Heuristic that is popularly used:
 - Reduce dimension through a supervised learning approach that predicts CTR using various user and item features for “exploit” phase
 - Explore by adding some randomization in an optimistic way
- Widely used supervised learning approach
 - Logistic Regression with smoothing, multi-hierarchy smoothing
- Exploration schemes
 - Epsilon-greedy, restricted epsilon-greedy, Thompson sampling, UCB

DATA



CONTEXT

Select Item j with item covariates \mathbf{z}_j
(keywords, content categories, ...)



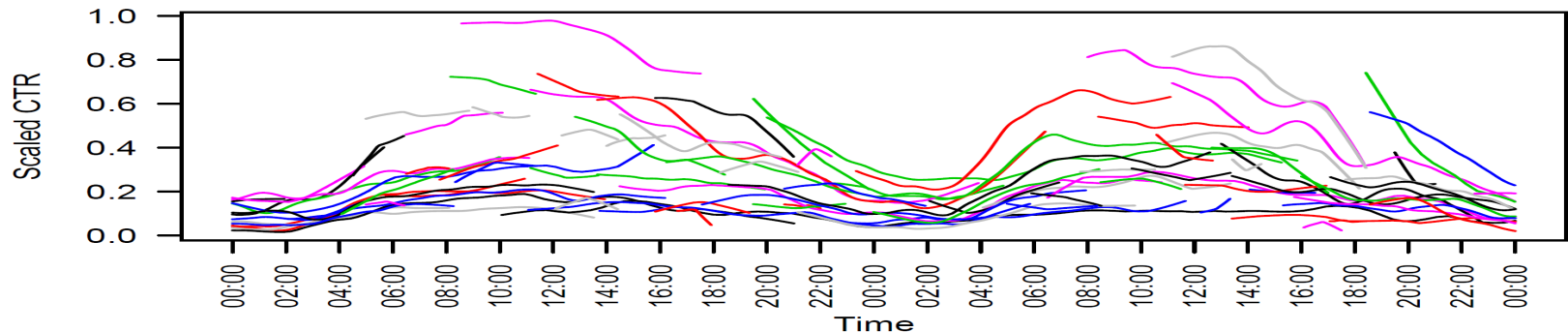
User i visits
(User, context)
covariates \mathbf{x}_{it}

(profile information, device id,
first degree connections,
browse information,...)

$(i, j) : \text{response } y_{ij}$
(click/no-click)

Illustrate with Y! front Page Application

- Simplify: Maximize CTR on first slot (F1)
- Article Pool
 - Editorially selected for high quality and brand image
 - Few articles in the pool but article pool dynamic



- Users with many prior visits see recommendations “tailored” to their taste, others see the best for the “group” they belong to

Types of user covariates

- Demographics, geo:
 - Not useful in front-page application
- Browse behavior: activity on Y! network (\mathbf{x}_{it})
 - Previous visits to property, search, ad views, clicks,...
 - This is useful for the front-page application
- Latent user factors based on previous clicks on the module (\mathbf{u}_i)
 - Useful for active module users, obtained via factor models(more later)
 - Teases out module affinity that is not captured through other user information, based on past user interactions with the module

Approach: Online + Offline

- Offline computation
 - Intensive computations done infrequently (once a day/week) to update parameters that are less time-sensitive
- Online computation
 - Lightweight computations frequent (once every 5-10 minutes) to update parameters that are time-sensitive
 - Exploration also done online

Online computation: per-item online logistic regression

- For item j , the state-space model is

$$\begin{aligned}y_{ijt} &\sim \text{Bernoulli}(p_{ijt}) \\ \text{logit}(p_{ijt}) &= X_i' A Z_j + \mathbf{u}_i' \mathbf{v}_{jt} + x_i' \boldsymbol{\beta}_{jt} \\ (\mathbf{v}_{j,t+1}, \boldsymbol{\beta}_{j,t+1}) &= (\mathbf{v}_{j,t}, \boldsymbol{\beta}_{j,t}) + \text{error}_{j,t+1} \\ (\mathbf{v}_{j,0}, \boldsymbol{\beta}_{j,0}) &= (0, 0) + \text{error}_{j,0}\end{aligned}$$

X_i : All user features (including latent factors)

A : Coefficients estimated offline through a big logistic regression

$(\mathbf{v}_{jt}, \boldsymbol{\beta}_{jt})$: Item specific coefficients updated online frequently

Item coefficients are update online via Kalman-filter

Explore/Exploit

- Three schemes (all work reasonably well for the front page application)
 - epsilon-greedy: Show article with maximum posterior mean except with a small probability epsilon, choose an article at random.
 - Upper confidence bound (UCB): Show article with maximum score, where score = post-mean + k. post-std
 - Thompson sampling: Draw a sample ($\mathbf{v}, \boldsymbol{\beta}$) from posterior to compute article CTR and show article with maximum drawn CTR

Computing the user latent factors(the u's)

- Computing user latent factors
 - This is computed offline once a day using retrospective (user,item) interaction data for last X days (X = 30 in our case)
 - Computations are done on Hadoop

Regression based Latent Factor Model

$y_{ij} \sim \text{Ber}(p_{ij})$ (# obs. per user has wide variation)

$$\text{lg} t(p_{ij}) = \hat{a}_{ik} u_{ik} v_{jk} = \mathbf{u}_i^T \mathbf{v}_j \quad (\text{need shrinkage on factors})$$

$$\mathbf{u}_i = \mathbf{G} \mathbf{x}_i + e_i^u, \quad e_i^u \sim N(0, \text{diag}(S_1^2, S_2^2, \dots, S_r^2))$$

regression weight matrix

user/item-specific correction terms (learnt from data)

$$\mathbf{v}_j = \mathbf{D} \mathbf{Z}_j + e_j^v, \quad e_j^v \sim N(0, I)$$

$$v_{ik} \quad 3 \quad 0$$

Role of shrinkage (consider Guassian for simplicity)

- For new user/article, factor estimates based on covariates

$$\mathbf{u}_{new} = \hat{G} \mathbf{x}_{new}^{user}, \quad \mathbf{v}_{new} = \hat{D} \mathbf{z}_{new}^{item}$$

For old user, factor estimates

$$E(\mathbf{u}_i | \text{Rest}) = \left(I + \sum_{j \in N_i} \hat{\alpha} \mathbf{v}_j \mathbf{v}_j' \right)^{-1} \left(\hat{G} \mathbf{x}_i^{user} + \sum_{j \in N_i} \hat{\alpha} y_{ij} \mathbf{v}_j \right)$$

- Linear combination of prior regression function and user feedback on items

Estimating the Regression function via EM

Maximize

$$\int \left(\prod_{ij} f(\mathbf{u}_i, \mathbf{v}_j, Data) \prod_i g(\mathbf{u}_i, G) \prod_j g(\mathbf{v}_j, D) \right) \prod_i d\mathbf{u}_i \prod_j d\mathbf{v}_j$$

Integral cannot be computed in closed form,
approximated by Monte Carlo using Gibbs Sampling

For logistic, we use ARS (Gilks and Wild) to sample the latent factors within the Gibbs sampler

Scaling to large data on via distributed computing (e.g. Hadoop)

- Randomly partition by users
- Run separate model on each partition
 - Care taken to initialize each partition model with same values, constraints on factors ensure “identifiability of parameters” within each partition
- Create ensembles by using different user partitions, average across ensembles to obtain estimates of user factors and regression functions
 - Estimates of user factors in ensembles uncorrelated, averaging reduces variance

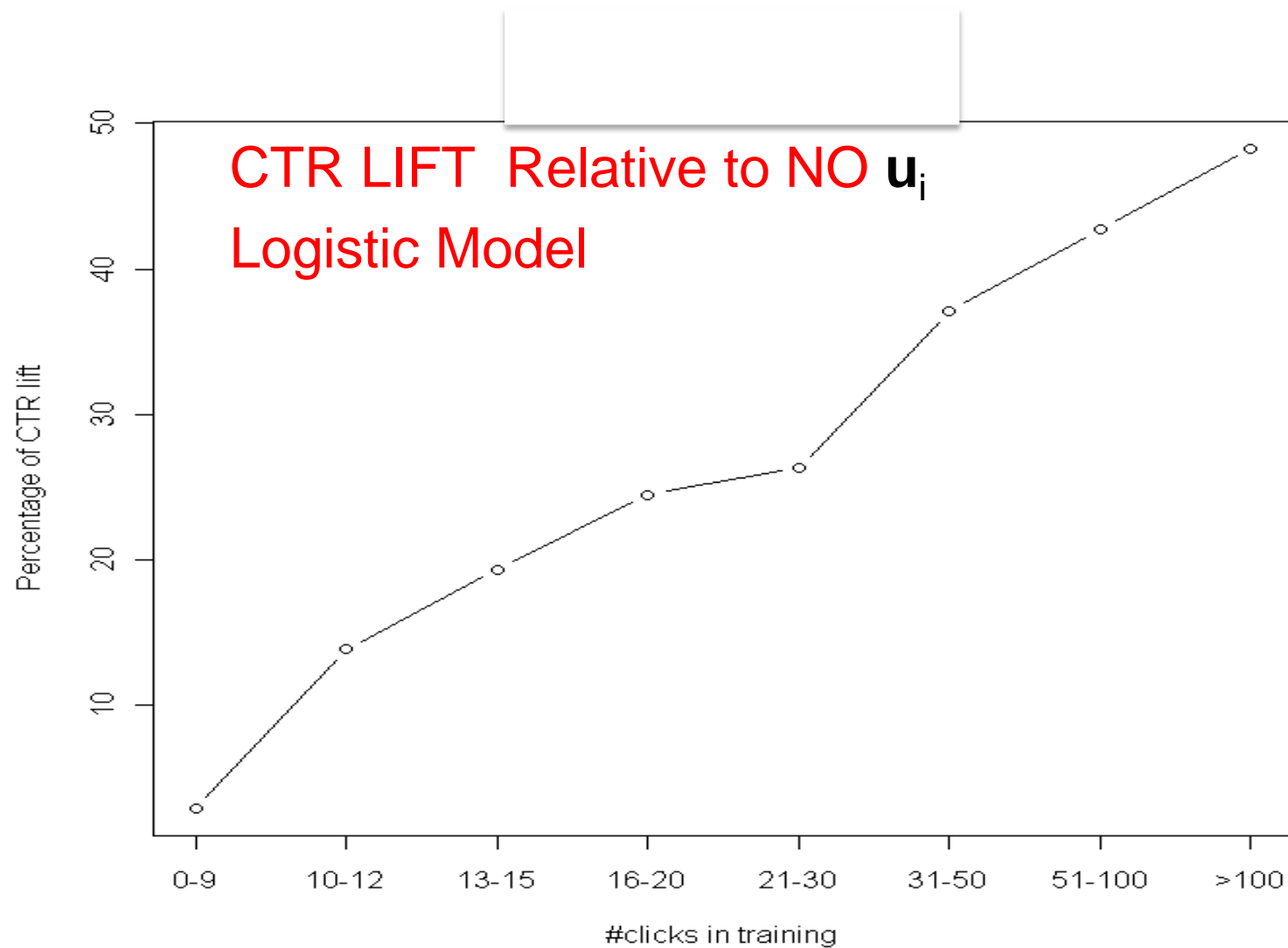
Data Example

- 1B events, 8M users, 6K articles
- Offline training produced user factor \mathbf{u}_i
- Our Baseline: logistic without user feature \mathbf{u}_i

$$\text{logit}(p_{ijt}) = x_{it}' b_{jt}$$

- Overall click lift by including \mathbf{u}_i : 9.7%,
- Heavy users (> 10 clicks last month): 26%
- Cold users (not seen in the past): 3%

Click-lift for heavy users



Multiple Objectives: An example in Content Optimization

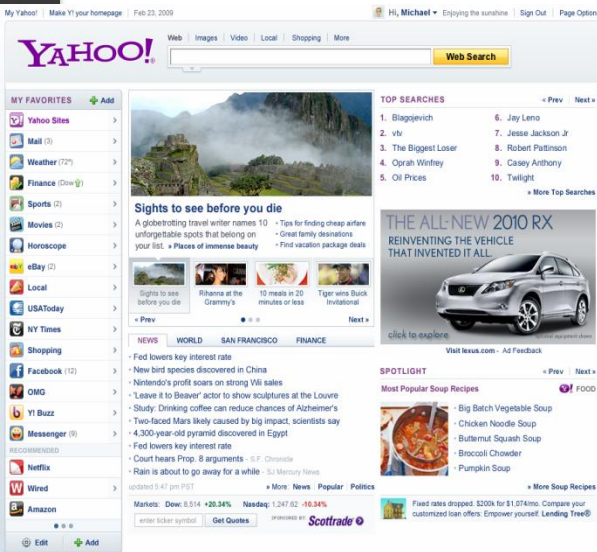
Recommender



EDITORIAL

content

Clicks on FP links influence
downstream supply distribution



SPORTS

NEWS

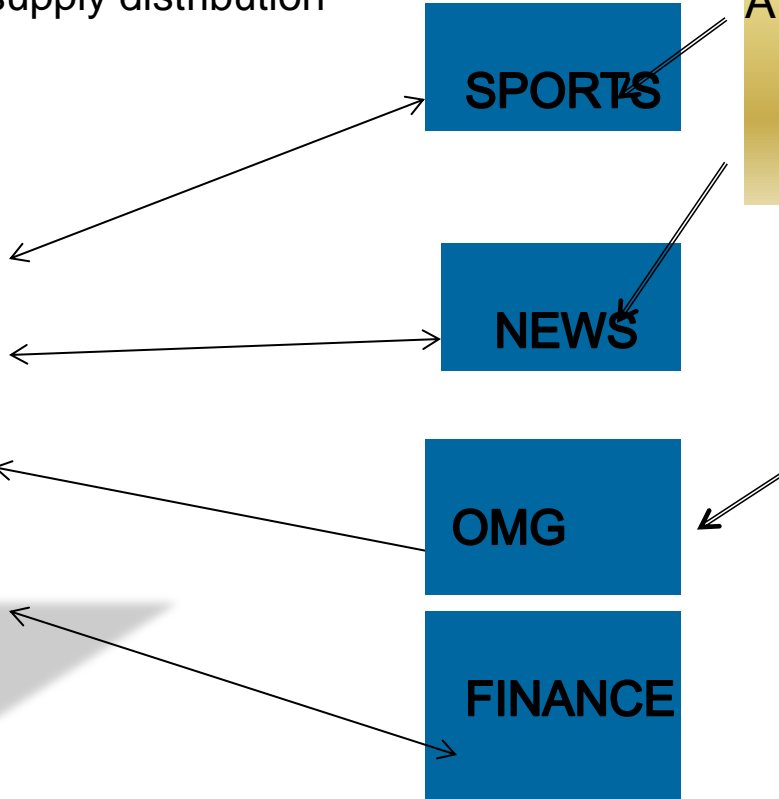
OMG

FINANCE

AD SERVER

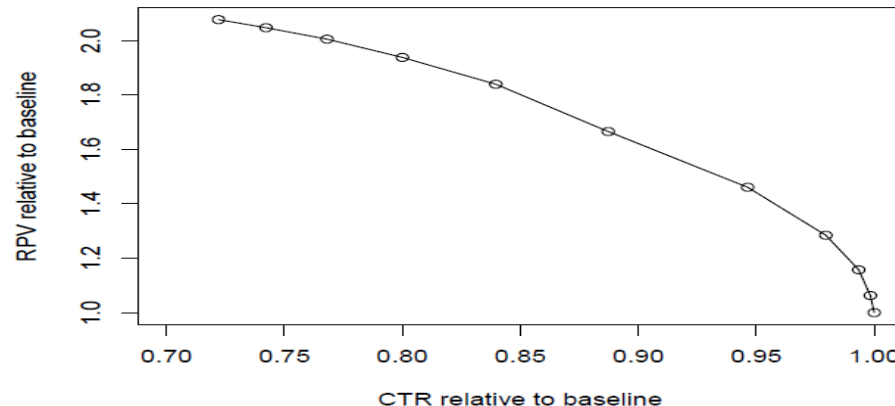
DISPLAY
ADVERTISING Revenue

Downstream
engagement
(Time spent)



Multiple Objectives

- What do we want to optimize?
- One objective: Maximize clicks
- But consider the following
 - Article 1: CTR=5%, utility per click = 5
 - Article 2: CTR=4.9%, utility per click=10
 - By promoting 2, we lose 1 click/100 visits, gain 5 utils
- If we do this for a large number of visits --- lose some clicks but obtain significant gains in utility?
 - E.g. lose 5% relative CTR, gain 40% in utility (e.g revenue, time spent)



An example of Multi-Objective Optimization

(Details: Agarwal et al, SIGIR 2012)

w_{ij} : probability of serving item j to user i

d_{ij} : downstream utility, p_{ij}^* : max CTR item

$$\max \sum_i \sum_j w_{ij} p_{ij} d_{ij} + \text{small strong convexity}$$

$$\text{s.t.} (1 - (\sum_i \sum_j w_{ij} p_{ij} / \sum_i p_{ij}^*)) < \alpha$$

Solution: Items ranked by $p_{ij} d_{ij} + \mu p_{ij}$

 Lagrange multiplier

LinkedIn Advertising: Brand, Self-Serve, Sponsored updates

The screenshot shows a LinkedIn profile for Liang Zhang. The page layout includes a top navigation bar, a main feed of updates, and a right-hand sidebar with recommendations and network statistics.

Updates and Advertisements:

- LinkedIn Today** recommends this news for you: **T. Boone Pickens: Why is Energy a Debate Taboo?** (linkedin.com)
- Top Stories:** How Microsoft is Going After Spotify, What to Do About Bullying Bosses, More (linkedin.com)
- Get Insights and Ideas Shared by Thought Leaders** (See more »)
- Andrea LaCoy** is now connected to **Mike Plumpe**, Principal Development Manager at Microsoft, **Peggy Crowley**, Senior Paralegal at Microsoft, **Roi Blanco**, Research Scientist at Yahoo! and 7 other people. (Send a message • 1 minute ago)
- Citi** Winners of the FT/Citi Ingenuity Awards, decided by an esteemed judging panel, will be announced on Dec. 5. In the meantime, we'll be highlighting all of the finalists across each of the 4 categories - energy, infrastructure, healthcare, and education. Stay tuned for updates! (Like • Comment • Share • 3 minutes ago)
- Yufei Li** is now connected to **Mengqiu Wang**, Software Engineer at Twitter. (Send a message • 10 minutes ago)
- Bin Liu** is now connected to **Chao Chen**, Graduate student at Northeastern University. (Send a message • 10 minutes ago)

Advertisements (highlighted with orange boxes):

- Alteryx Free Trial** - Create Your Own BI Workflows. Try Alteryx Free For 30-Days!
- HTML5 Sencha PhoneGap** For mobile app dev, learn when to use HTML5 vs. Native vs. Hybrid
- Dislike Cold Calls?** Take a free trial of ZoomInfo today and take the chill out of cold calls.

PEOPLE YOU MAY KNOW

- Zhibin Yuan**, Deputy Director (attachment) of Division of [Connect](#)
- Nan Zhang**, ERP BASIS consultant at PetroChina [Connect](#)
- Brad Malin**, Associate Professor at Vanderbilt University and [Connect](#)

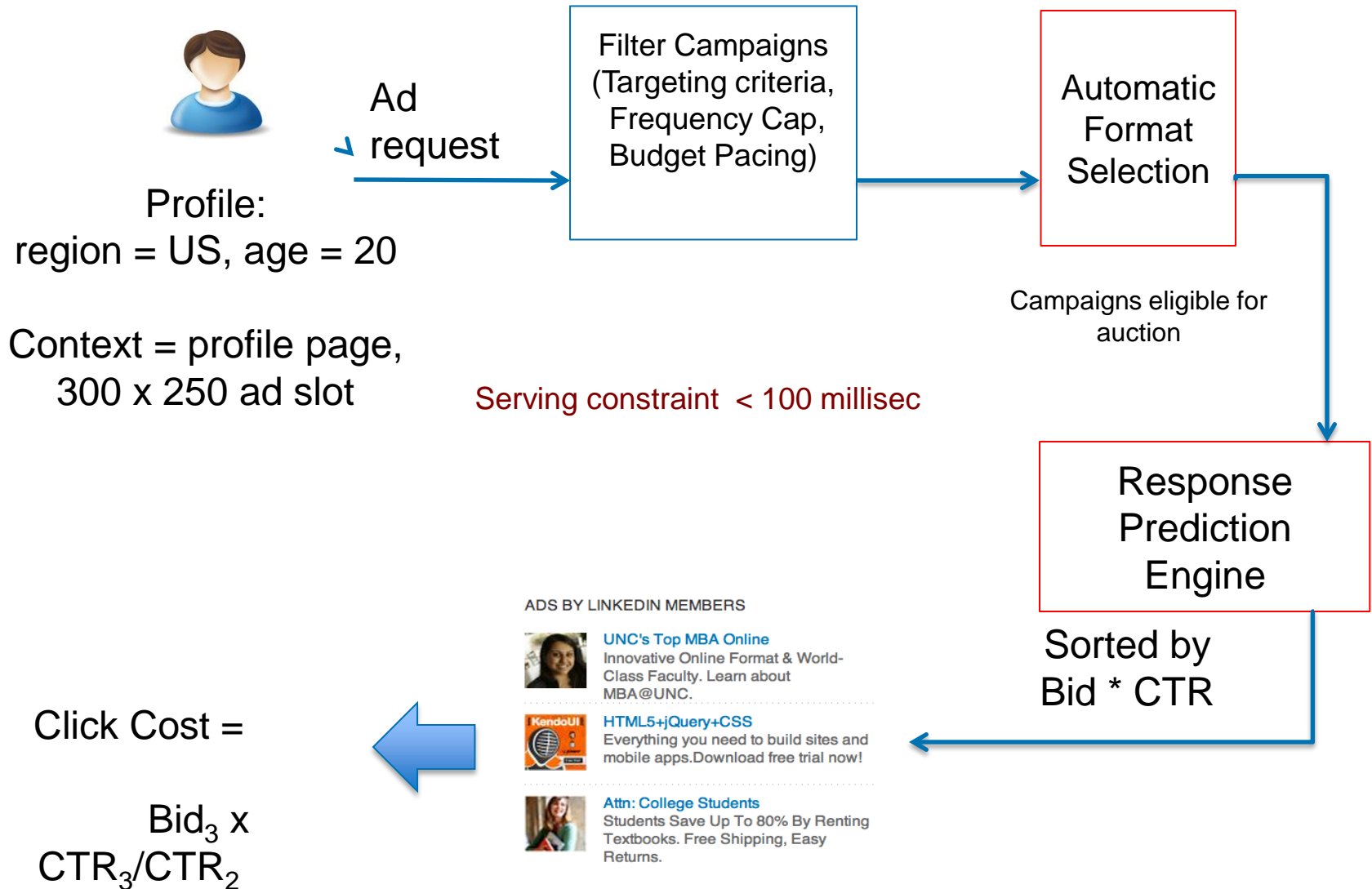
WHO'S VIEWED YOUR PROFILE?

- 25** Your profile has been viewed by 25 people in the past 7 days.
- 15** You have shown up in search results 15 times in the past 3 days.

YOUR LINKEDIN NETWORK

- 198** Connections link you to 4,817,346+ professionals
- 28,520** New people in your Network since October 11

SERVING



CTR Prediction Model for Ads

- Feature vectors
 - Member feature vector: x_i
 - Campaign feature vector: c_j
 - Context feature vector: z_k

- Model:

$$y_{ijt} \sim \text{Bernoulli}(p_{ijt}),$$

$$p_{ijt} = \frac{1}{1 + \exp(-s_{ijt})}.$$

$$s_{ijt} = \omega + s_{ijt}^{1,c} + s_{ijt}^{2,c} + s_{ijt}^{2,w}$$

$$s_{ijt}^{1,c} = x_i' \alpha + c_j' \beta + z_t' \gamma$$

$$s_{ijt}^{2,c} = x_i' A c_j + x_i' C z_t + z_t' B c_j$$

$$s_{ijt}^{2,w} = \delta_j + x_i' \eta_j + z_t' \xi_j$$

CTR Prediction Model for Ads

- Feature vectors
 - Member feature vector: x_i
 - Campaign feature vector: c_j
 - Context feature vector: z_k

- Model:

$$y_{ijt} \sim \text{Bernoulli}(p_{ijt}),$$

$$p_{ijt} = \frac{1}{1 + \exp(-s_{ijt})}.$$

$$s_{ijt} = \omega + s_{ijt}^{1,c} + s_{ijt}^{2,c} + s_{ijt}^{2,w}$$

$$s_{ijt}^{1,c} = x_i' \alpha + c_j' \beta + z_t' \gamma$$

$$s_{ijt}^{2,c} = x_i' A c_j + x_i' C z_t + z_t' B c_j$$

$$s_{ijt}^{2,w} = \delta_j + x_i' \eta_j + z_t' \xi_j$$

Cold-start component

Warm-start
per-campaign component

CTR Prediction Model for Ads

- Feature vectors

- Member feature vector: x_i
- Campaign feature vector: c_j
- Context feature vector: z_k

- Model:

$$y_{ijt} \sim \text{Bernoulli}(p_{ijt}),$$

$$p_{ijt} = \frac{1}{1 + \exp(-s_{ijt})}.$$

$$s_{ijt} = \omega + s_{ijt}^{1,c} + s_{ijt}^{2,c} + s_{ijt}^{2,w}$$

$$s_{ijt}^{1,c} = x_i' \alpha + c_j' \beta + z_t' \gamma$$

$$s_{ijt}^{2,c} = x_i' A c_j + x_i' C z_t + z_t' B c_j$$

$$s_{ijt}^{2,w} = \delta_j + x_i' \eta_j + z_t' \xi_j$$

Cold-start:

$$\Theta_c = \{\omega, \alpha, \beta, \gamma, A, B, C\}$$

Warm-start:

$$\Theta_w = \{\delta_j, \eta_j, \xi_j\}, j = 1, \dots, J.$$

Both can have L2 penalties.

Cold-start component

Warm-start
per-campaign component

Model Fitting

- Single machine (well understood)
 - conjugate gradient
 - L-BFGS
 - Trusted region
 - ...
- Model Training with Large scale data
 - Cold-start component Θ_w is more stable
 - Weekly/bi-weekly training good enough
 - However: difficulty from need for large-scale logistic regression
 - Warm-start per-campaign model Θ_c is more dynamic
 - New items can get generated any time
 - Big loss if opportunities missed
 - Need to update the warm-start component as frequently as possible

Model Fitting

- Single machine (well understood)

- conjugate gradient
- L-BFGS
- Trusted region
- ...

- Model Training with Large scale data

- Cold-start component Θ_w is more stable
 - Weekly/bi-weekly training good enough
 - However: difficulty from need for large-scale logistic regression
- Warm-start per-campaign model Θ_c is more dynamic
 - New items can get generated any time
 - Big loss if opportunities missed
 - Need to update the warm-start component as frequently as possible

Large Scale Logistic Regression



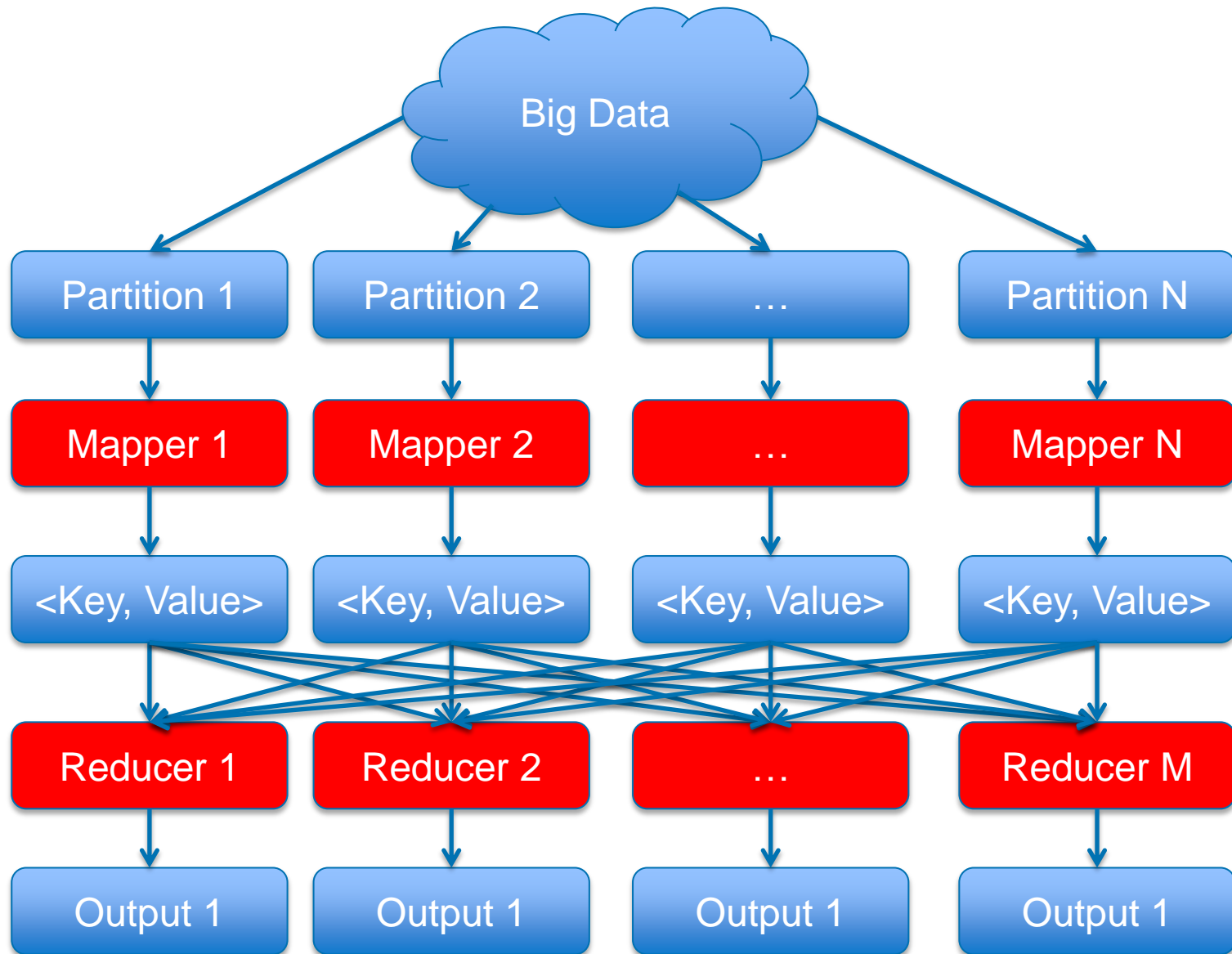
Per-item logistic regression given Θ_c



Large Scale Logistic Regression: Computational Challenge

- Hundreds of millions/billions of observations
- Hundreds of thousands/millions of covariates
- Fitting a logistic regression model on a single machine not feasible
- Model fitting iterative using methods like gradient descent, Newton's method etc
 - Multiple passes over the data
- Problem: Find x to $\min(F(x))$
- Iteration n : $x_n = x_{n-1} - b_{n-1} F'(x_{n-1})$
- b_{n-1} is the step size that can change every iteration
- Iterate until convergence
- Conjugate gradient, LBFGS, Newton trust region, ...

Compute using Map-Reduce

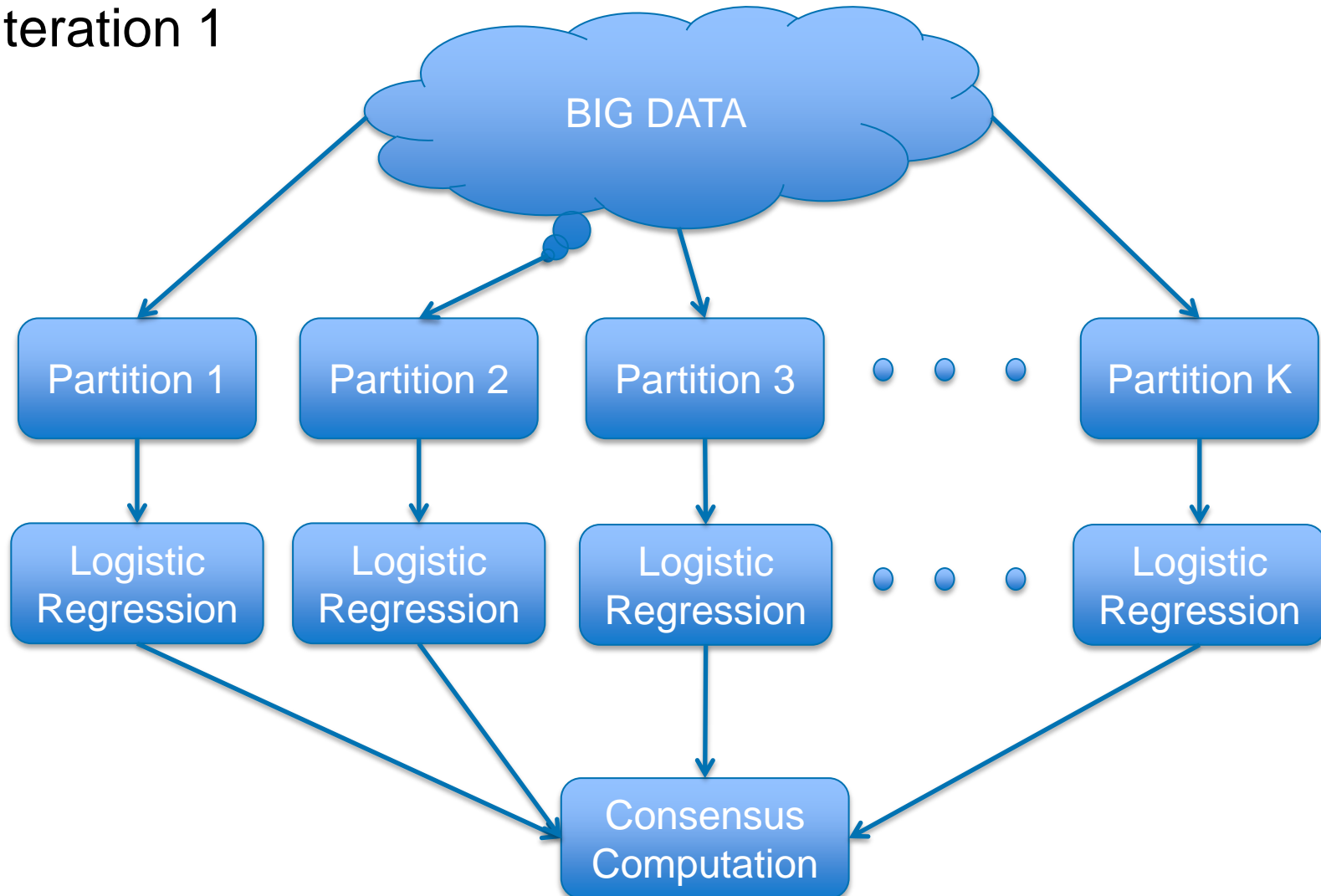


Large Scale Logistic Regression

- Naïve:
 - Partition the data and run logistic regression for each partition
 - Take the mean of the learned coefficients
 - Problem: Not guaranteed to converge to global solution
- Alternating Direction Method of Multipliers (ADMM)
 - Boyd et al. 2011
 - Set up constraints: each partition's coefficient = global consensus
 - Solve the optimization problem using Lagrange Multipliers
 - Advantage: converges to global solution

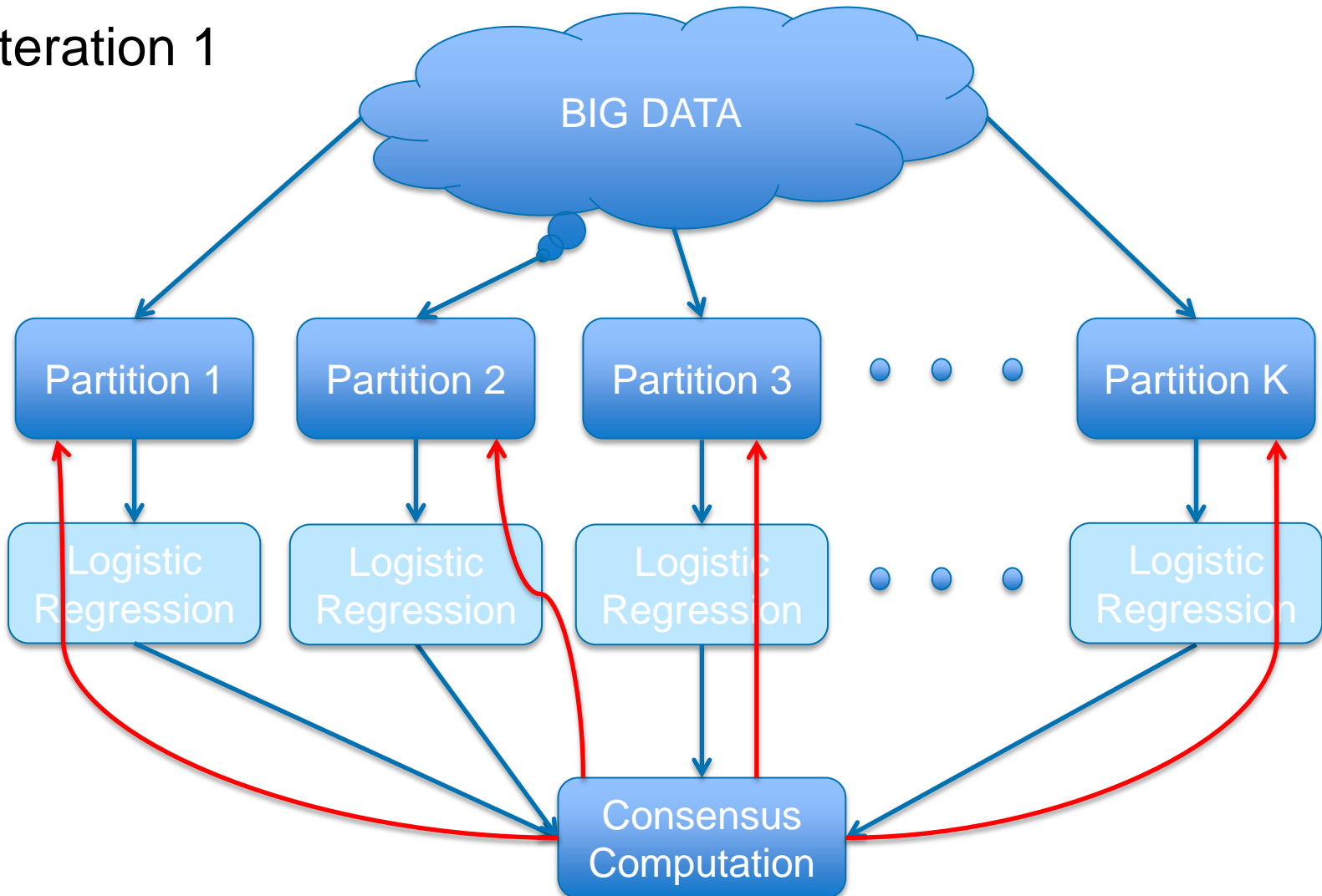
Large Scale Logistic Regression via ADMM

Iteration 1



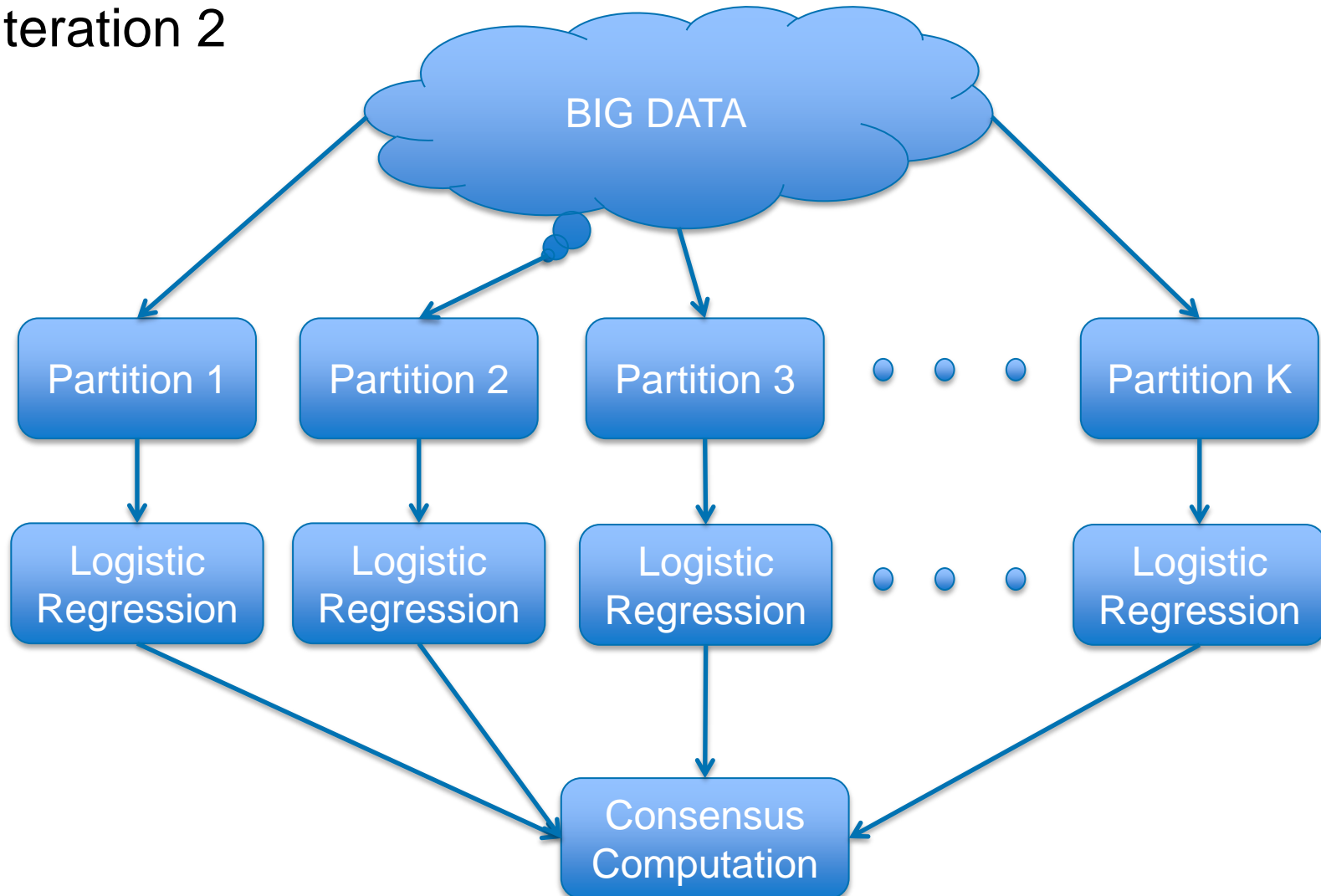
Large Scale Logistic Regression via ADMM

Iteration 1



Large Scale Logistic Regression via ADMM

Iteration 2



Large Scale Logistic Regression via ADMM

■ Notation

- $(\mathbf{X}_i, \mathbf{y}_i)$: data in the i^{th} partition
- $\boldsymbol{\beta}_i$: coefficient vector for partition i
- $\boldsymbol{\beta}$: Consensus coefficient vector
- $r(\boldsymbol{\beta})$: penalty component such as $\|\boldsymbol{\beta}\|_2^2$

■ Optimization problem

$$\min \sum_{i=1}^N l_i(\mathbf{y}_i, \mathbf{X}_i^T \boldsymbol{\beta}_i) + r(\boldsymbol{\beta})$$

subject to $\boldsymbol{\beta}_i = \boldsymbol{\beta}$

ADMM updates

$$\boldsymbol{\beta}_i^{k+1} = \operatorname{argmin}_{\boldsymbol{\beta}_i} (l_i(\mathbf{y}_i, \mathbf{X}_i^T \boldsymbol{\beta}_i)$$

$$+ \frac{\rho}{2} \|\boldsymbol{\beta}_i - \boldsymbol{\beta}^k + \mathbf{u}_i^k\|_2^2)$$

LOCAL REGRESSIONS
Shrinkage towards current
best global estimate

$$\boldsymbol{\beta}^{k+1} = \operatorname{argmin}_{\boldsymbol{\beta}} (r(\boldsymbol{\beta})$$

$$+ \frac{N\rho}{2} \|\boldsymbol{\beta} - \bar{\boldsymbol{\beta}}^{k+1} - \bar{\mathbf{u}}^k\|_2^2)$$

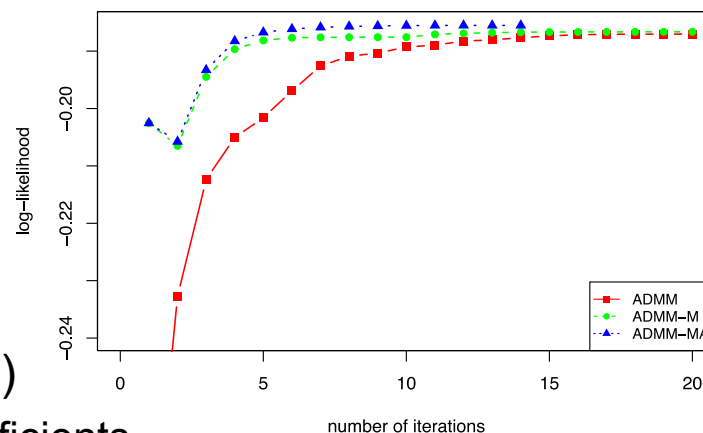
UPDATED
CONSENSUS

$$\mathbf{u}_i^{k+1} = \mathbf{u}_i^k + \boldsymbol{\beta}_i^{k+1} - \boldsymbol{\beta}^{k+1}$$

ADMM at LinkedIn

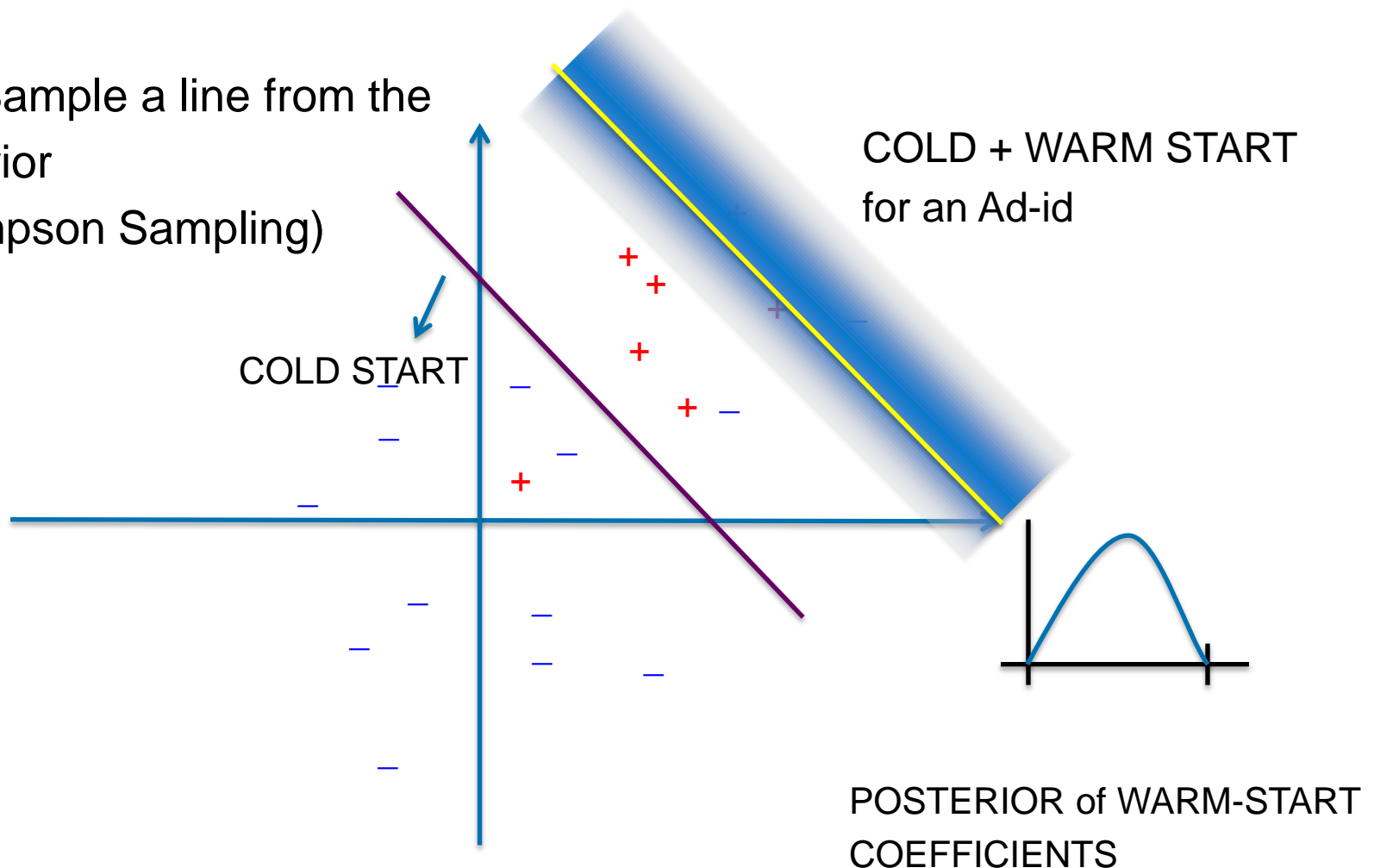
- Lessons and Improvements

- Initialization is important (ADMM-M)
 - Use the mean of the partitions' coefficients
 - Reduces number of iterations by 50%
- Adaptive step size (learning rate) (ADMM-MA)
 - Exponential decay of learning rate
- Together, these optimizations reduce training time from 10h to 2h



Explore/Exploit with Logistic Regression

E/E: Sample a line from the posterior
(Thompson Sampling)



Models Considered

- CONTROL: per-campaign CTR counting model
- COLD-ONLY: only cold-start component
- LASER: our model (cold-start + warm-start)
- LASER-EE: our model with Explore-Exploit using Thompson sampling

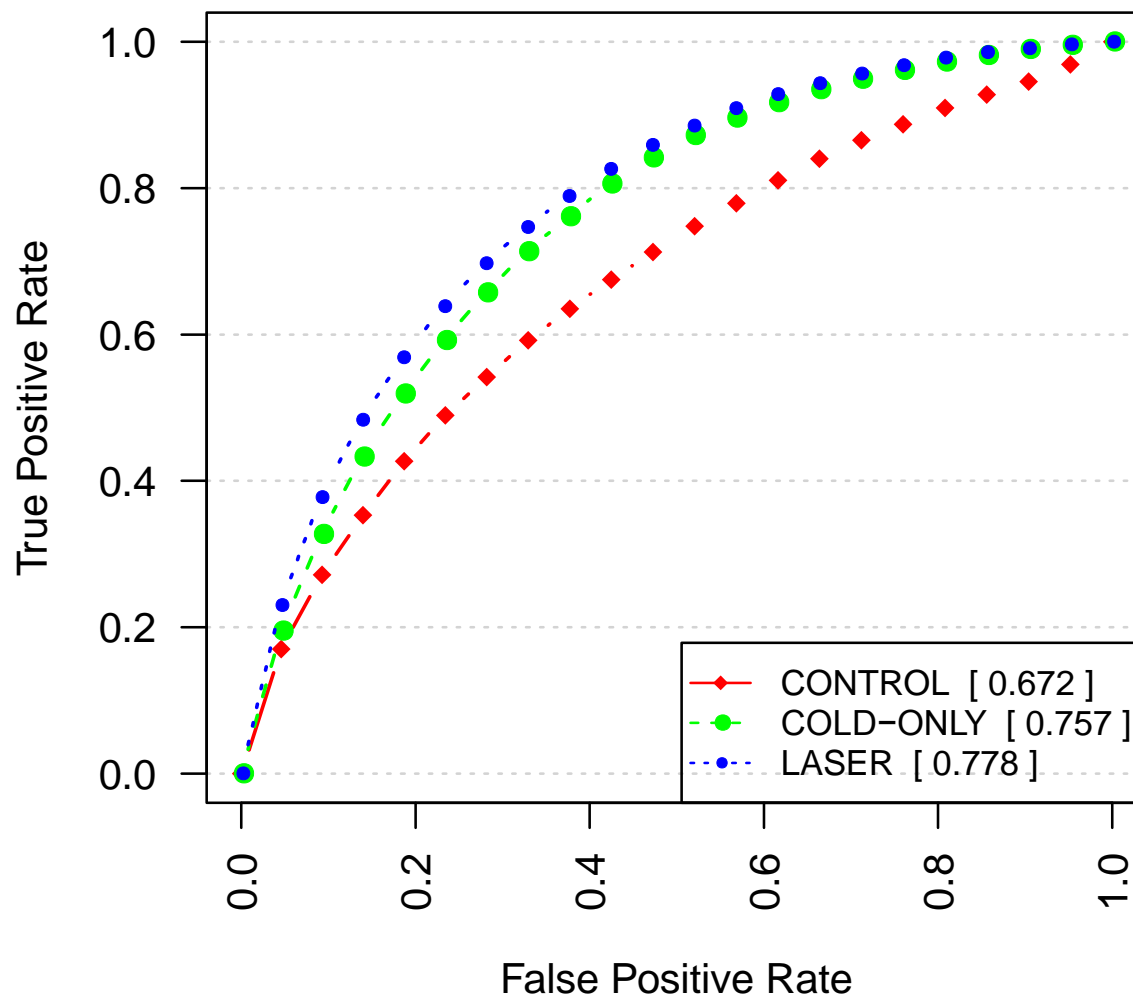
Metrics

- Model metrics
 - Test Log-likelihood
 - AUC/ROC
 - Observed/Expected ratio
- Business metrics (Online A/B Test)
 - CTR
 - CPM (Revenue per impression)

Observed / Expected Ratio

- Observed: #Clicks in the data
- Expected: Sum of predicted CTR for all impressions
- Not a “standard” classifier metric, but in many ways more useful for this application
- What we usually see: Observed / Expected < 1
 - Quantifies the “winner’s curse” aka selection bias in auctions
 - When choosing from among thousands of candidates, an item with mistakenly over-estimated CTR may end up winning the auction
- Particularly helpful in spotting inefficiencies by segment
 - E.g. by bid, number of impressions in training (warmness), geo, etc.
 - Allows us to see where the model might be giving too much weight to the wrong campaigns
- High correlation between O/E ratio and model performance online

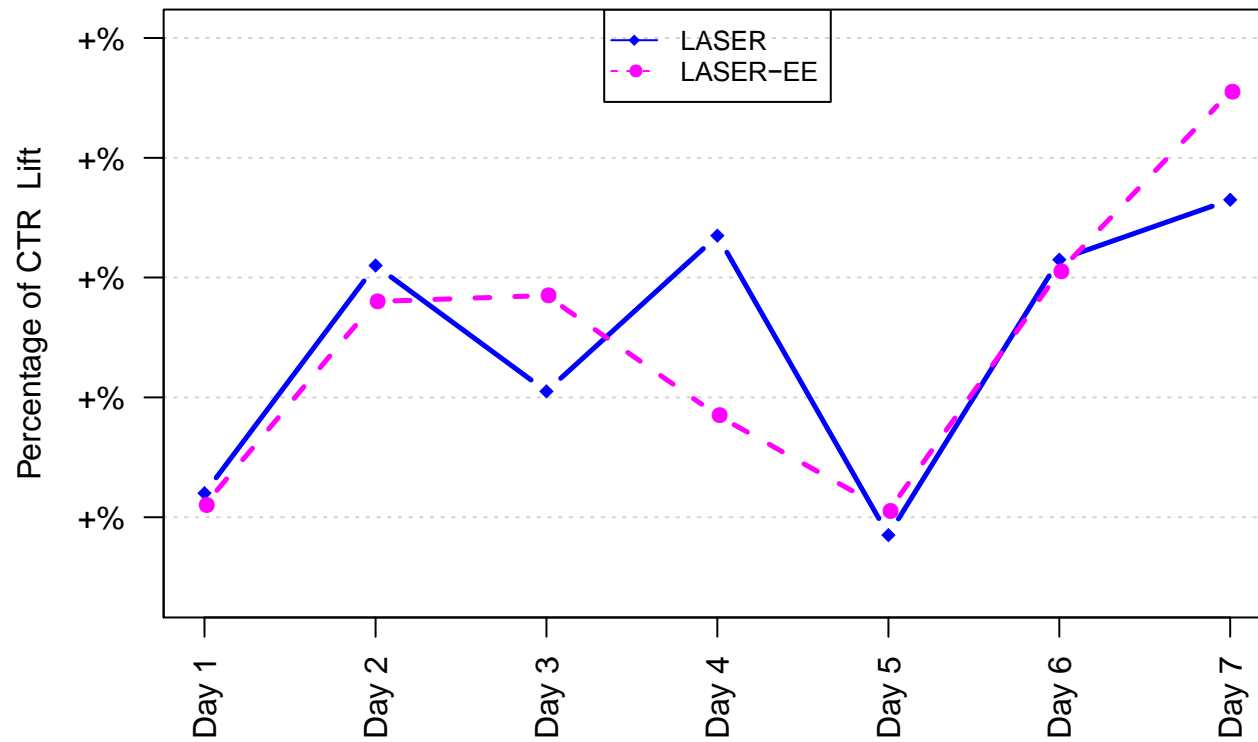
Offline: ROC Curves



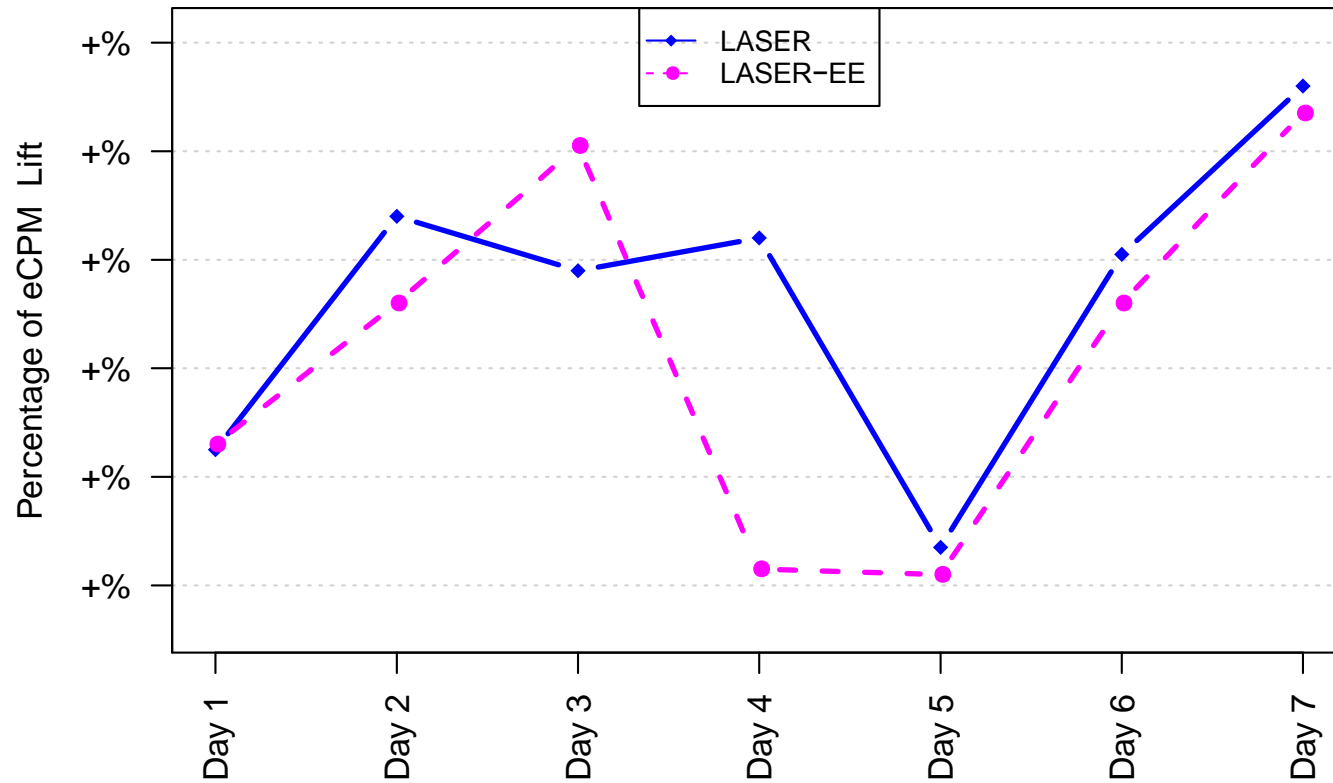
Online A/B Test

- Three models
 - CONTROL (10%)
 - LASER (85%)
 - LASER-EE (5%)
- Segmented Analysis
 - 8 segments by campaign warmness
 - Degree of warmness: the number of training samples available in the training data for the campaign
 - Segment #1: Campaigns with almost no data in training
 - Segment #8: Campaigns that are served most heavily in the previous batches so that their CTR estimate can be quite accurate

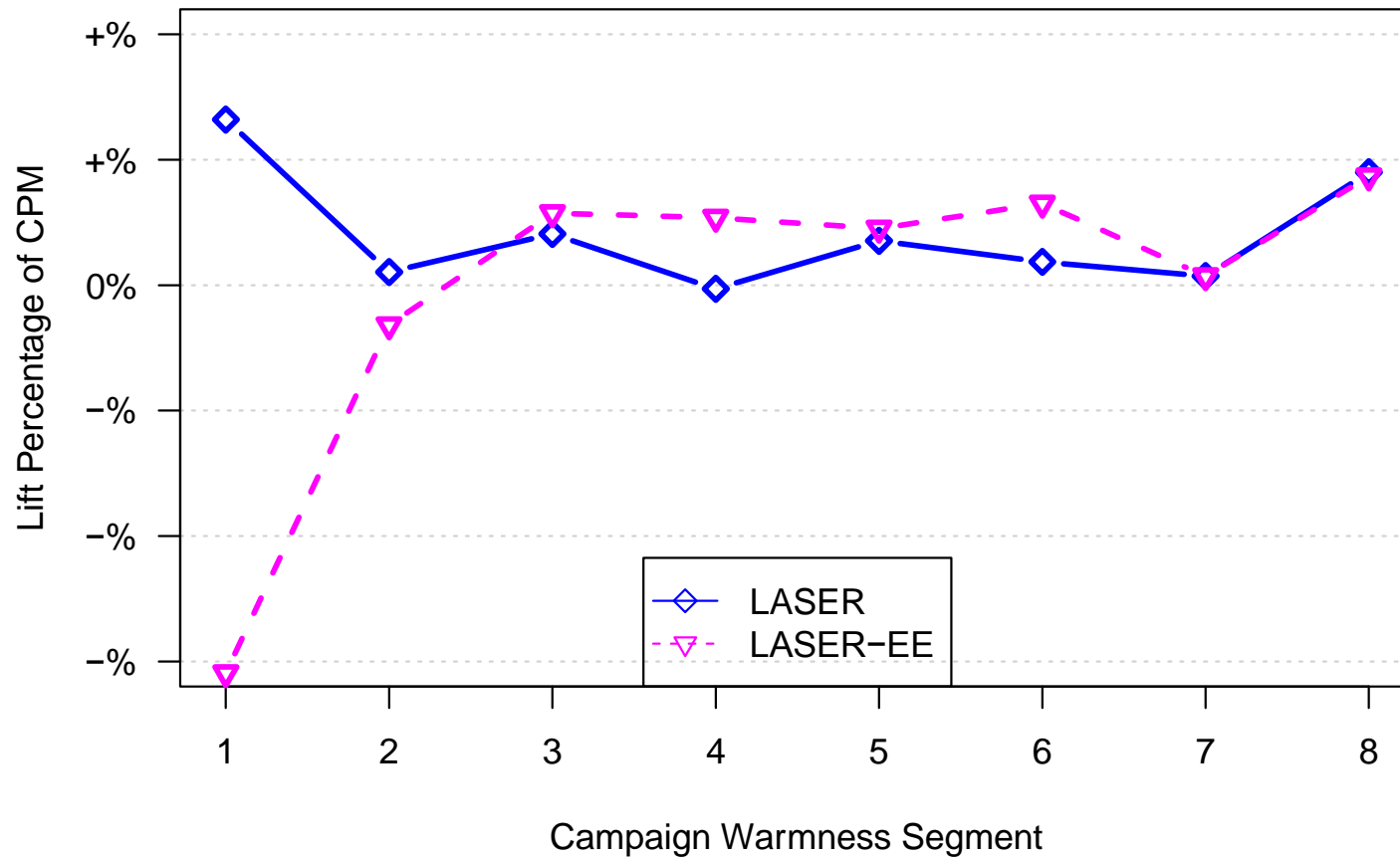
Daily CTR Lift Over Control



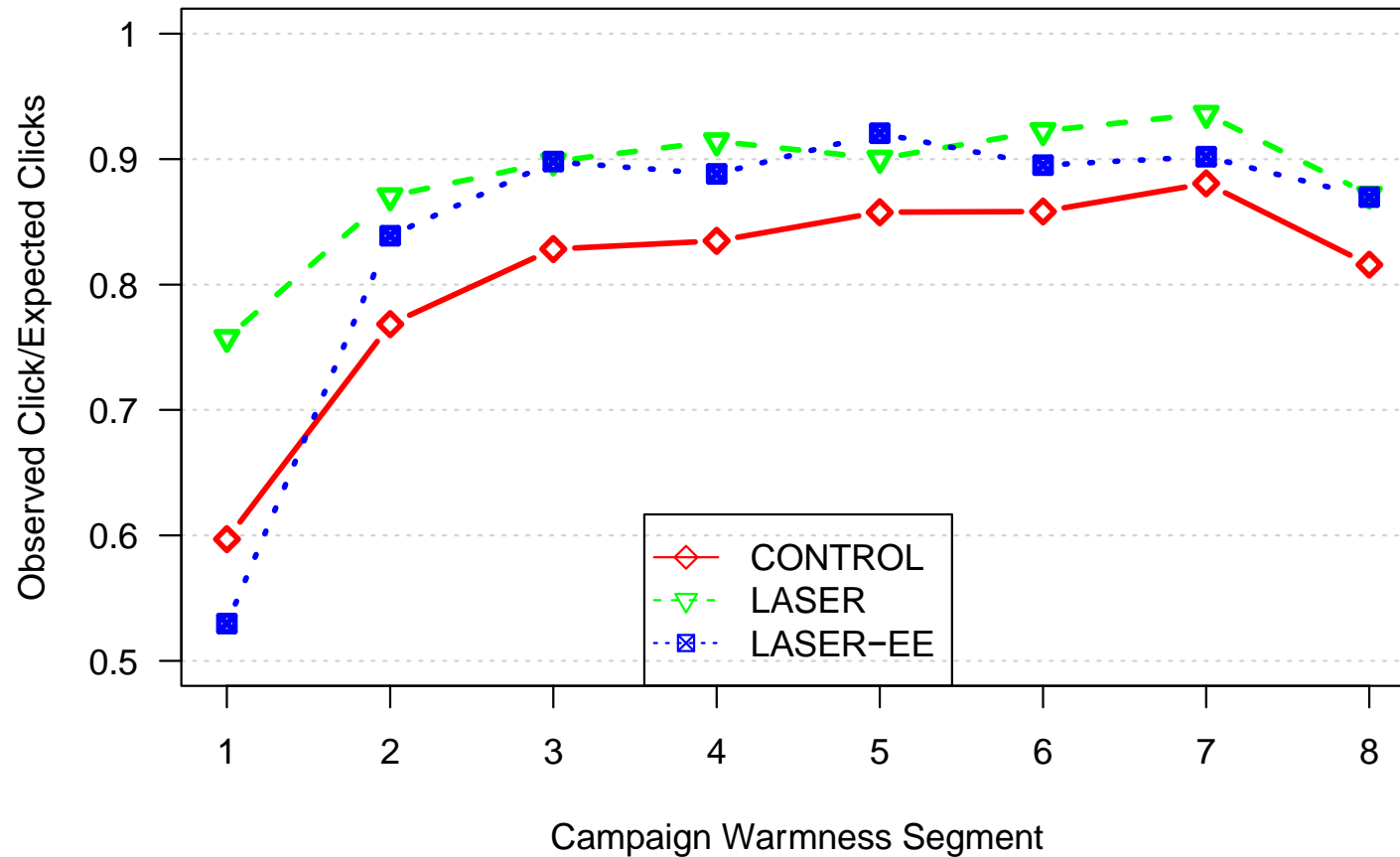
Daily CPM Lift Over Control



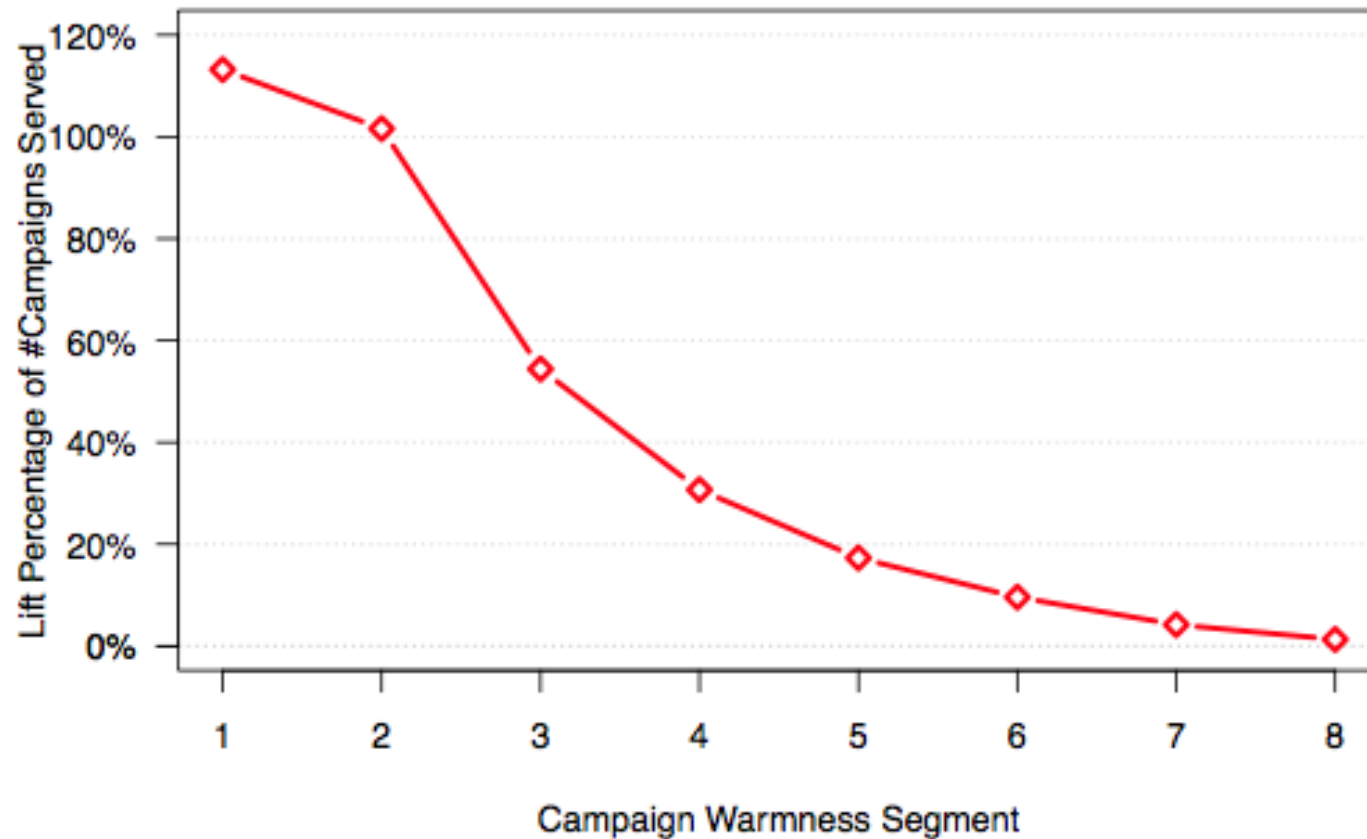
CPM Lift By Campaign Warmness Segments



O/E Ratio By Campaign Warmness Segments



Number of Campaigns Served Improvement from E/E



Insights

- Overall performance:
 - LASER and LASER-EE are both much better than control
 - LASER and LASER-EE performance are very similar
- Segmented analysis by campaign warmness
 - Segment #1 (very cold)
 - LASER-EE much worse than LASER due to its exploration property
 - LASER much better than CONTROL due to cold-start features
 - Segments #3 - #5
 - LASER-EE significantly better than LASER
 - Winner's curse hit LASER
 - Segment #6 - #8 (very warm)
 - LASER-EE and LASER are equivalent
- Number of campaigns served
 - LASER-EE serves significantly more campaigns than LASER
 - Provides healthier market place

Takeaways

- Reducing dimension through logistic regression coupled with explore/exploit schemes like Thompson sampling effective mechanism to solve response prediction problems in advertising
- Partitioning model components by cold-start (stable) and warm-start (non-stationary) with different training frequencies effective mechanism to scale computations
- ADMM with few modifications effective model training strategy for large data with high dimensionality
- Methods work well for LinkedIn advertising, significant improvements

Current Work

- Investigating Spark and various other fitting algorithms
 - Promising results, ADMM still looks good on our datasets
- Stream Ads
 - Multi-response prediction (clicks, shares, likes, comments)
 - Filtering low quality ads extremely important
 - Revenue/Engagement tradeoffs (Pareto optimal solutions)
- Stream Recommendation
 - Holistic solution to both content and ads on the stream
- Large scale ML infrastructure at LinkedIn
 - Powers several recommendation systems

Summary

- Large scale Machine Learning plays an important role in recommender problems
- Several such problems can be cast as explore/exploit tradeoff
- Estimating interactions in high-dimensional sparse data via supervised learning important for efficient exploration and exploitation
- Scaling such models to Big Data is a challenging statistical problem
- Combining offline + online modeling with classical explore/exploit algorithm is a good practical strategy

Other challenges

- 3Ms: Multi-response, Multi-context modeling to optimize Multiple Objectives
 - Multi-response: Clicks, share, comments, likes,.. (preliminary work at CIKM 2012)
 - Multi-context: Mobile, Desktop, Email,..(preliminary work at SIGKDD 2011)
 - Multi-objective: Tradeoff in engagement, revenue, viral activities
 - Preliminary work at SIGIR 2012, SIGKDD 2011
- Scaling model computations at run-time to avoid latency issues
 - Predictive Indexing (preliminary work at WSDM 2012)

Backup slides

LASER Configuration

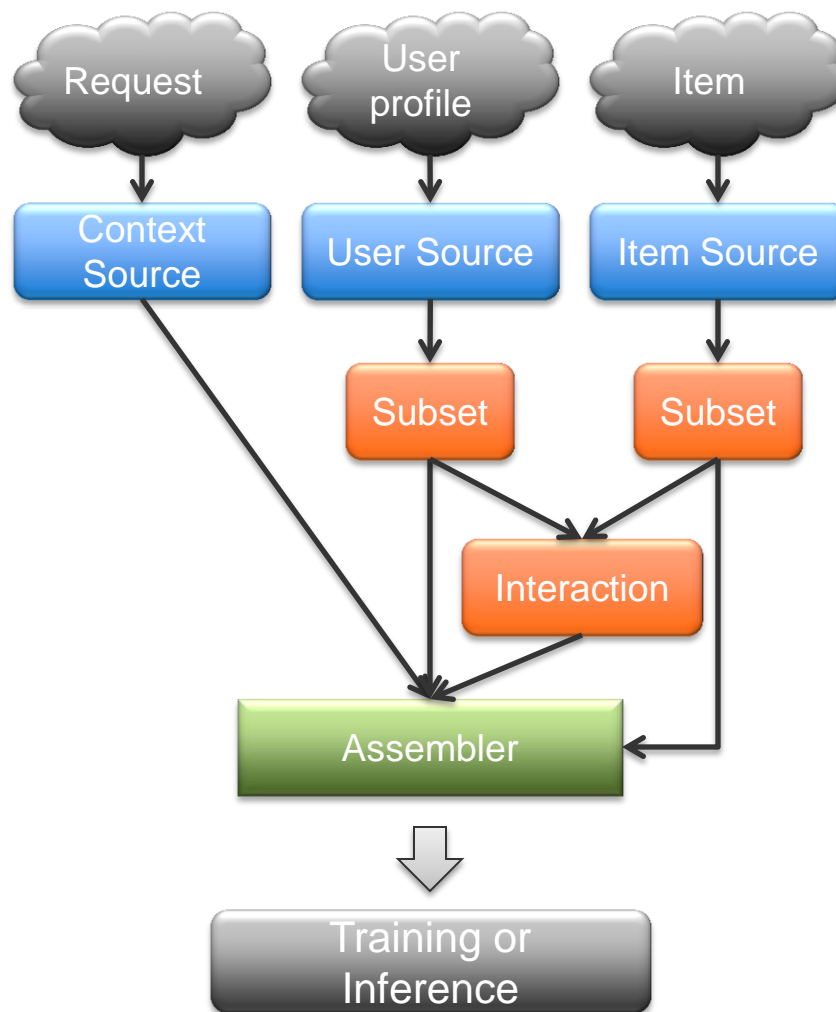
- Feature processing pipeline

- **Sources**: transform external data into feature vectors
- **Transformers**: modify/combine feature vectors
- **Assembler**: Packages features vectors for training/inference

- Configuration language

- Model structure can be changed extensively
- Library of reusable components
- Train, test, and deploy models without any code changes
- Speeds up model development cycle

LASER Transformer Pipeline



LASER Performance

- Real time inference
 - About 10 μ s per inference (1500 ads = 15ms)
 - Reacts to changing features immediately
- “Better wrong than late”
 - If a feature isn’t immediately available, back off to prior value
- Asynchronous computation
 - Actions that block or take time run in background threads
- Lazy evaluation
 - Sources & transformers do not create feature vectors for all items
 - Feature vectors are constructed/transformed only when needed
- Partial results cache
 - Logistic regression inference is a series of dot products
 - Scalars are small; cache can be huge
 - Hardware-like implementation to minimize locking and heap pressure

Summary

- Large scale Machine Learning plays an important role in computational advertising and content recommendation
- Several such problems can be cast as explore/exploit tradeoff
- Estimating interactions in high-dimensional sparse data via supervised learning important for efficient exploration and exploitation
- Scaling such models to Big Data is a challenging statistical problem
- Combining offline + online modeling with classical explore/exploit algorithm is a good practical strategy

Other challenges

- 3Ms: Multi-response, Multi-context modeling to optimize Multiple Objectives
 - Multi-response: Clicks, share, comments, likes,.. (preliminary work at CIKM 2012)
 - Multi-context: Mobile, Desktop, Email,..(preliminary work at SIGKDD 2011)
 - Multi-objective: Tradeoff in engagement, revenue, viral activities
 - Preliminary work at SIGIR 2012, SIGKDD 2011
- Scaling model computations at run-time to avoid latency issues
 - Predictive Indexing (preliminary work at WSDM 2012)