Contextual Bandits for Advertising Budget Allocation

Benjamin Han
Lyft
San Francisco, California
bhan@lyft.com

Jared Gabor
Lyft
San Francisco, California
jgabor@lyft.com

ABSTRACT

When allocating budgets across different ad campaigns, advertisers confront the challenge that the payouts or returns are uncertain. In this paper, we describe a system for optimizing advertising campaign budgets to ensure long-term profitability in the face of this uncertainty. Our modified contextual bandit system 1) applies supervised learning to predict ad campaign payouts based on context features and historical performance; 2) extrapolates the payouts to out-of-sample budgets using a simple functional form for the distribution of payouts; then 3) uses Thompson Sampling from the predicted payout distributions to manage the explore-exploit trade-off when selecting budgets. Using our system, we measure an overall efficiency improvement of (22 ± 10)% in the mean Cost Per Acquisition over the previous budget allocation strategy using Markov Chain Monte-Carlo. This system is now responsible for managing hundreds of millions of dollars of annual marketing spend at Lyft.

CCS CONCEPTS

• **Theory of computation** → **Bayesian analysis**; • **Information systems** → **Online advertising**; • **Computing methodologies** → *Reinforcement learning*.

KEYWORDS

Reinforcement Learning, Advertising, Contextual Bandit, Multi-Armed Bandits, Thompson Sampling, Semi-Supervised Learning, Bayesian Analysis, Portfolio Management

ACM Reference Format:
Benjamin Han and Jared Gabor. 2020. Contextual Bandits for Advertising Budget Allocation. In *The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 22–27, 2020, San Diego, CA, USA.* ACM, New York, NY, USA, 6 pages.

1 INTRODUCTION

Paid customer acquisition through advertising has spurred the growth of many firms. Yet the diversity of channels for customer acquisition raise a key problem: how to allocate budgets across many different advertising options. The challenges primarily stem from the uncertainty in the payouts for a given level of investment in a given advertising investment option.

KDD '20, August 22–27, 2020, San Diego, CA, USA

An advertiser might opt to use yesterday's performance naively to predict tomorrow's performance. This approach has three pitfalls: 1) past performance could be a poor predictor as the context – the seasonality, the competition for ad space, etc. – changes over time. 2) Even if the past is a good predictor, it might not be relevant if the advertising budget is far greater or lesser than what was spent before. 3) Past performance data are not available for new ads.

We present a multi-armed bandit system to overcome these challenges. We frame the performance prediction problem as a modified contextual bandit [10]. Our system leverages historical data and context information into a global model to make better predictions. The context information also enables reasonable predictions for new ads that share similarities with previous ads. The system combines the contextual predictions with economics principles to enable performance extrapolation to out-of-sample budgets. Through principled exploration of budgets using Thompson Sampling, the system collects diverse data to improve budget allocation over time [13].

This system, applied to driver acquisition at Lyft, more accurately predicts payouts, and produces cost efficiency gains of over 20% when compared to previous methods using Markov Chain Monte Carlo.

1.1 Background: Driver Acquisition at Lyft

Lyft provides a taxi-network for its users by dispatching drivers to ride requests. Lyft invests considerable time and effort to balance driver supply and rider demand. Some levers to optimize supply-demand balance have rapid responses (surge pricing mechanisms can suppress rider demand within minutes) while others have longer delays (driver notifications will not generate additional driver supply for hours). A slow but cost-efficient approach to manipulating supply and demand is user acquisition in the form of rider app installation and driver application. Our focus in this paper is a strategy to improve the cost-efficiency of driver acquisition through paid online advertising.

Driver acquisition advertising generates new drivers through targeted advertisements on ad platforms while respecting budgetary constraints. Our system actively manages roughly 10,000 marketing campaigns across 300 geographical regions by recommending a daily budget for each (campaign, region) tuple, accounting for hundreds of millions of dollars of acquisition spend annually.

To maintain reasonable scope in this paper, we simplify a few important elements of driver acquisition. First, we treat new drivers as interchangeable within a city, independent of work rate, day/week availability, vehicle, etc. Second, our system makes use of driver Lifetime Value (LTV) projections to ensure profitable investment. We take mean LTV per new driver (by region) as an input to the system. Third, a prospective driver may take weeks or months from the day they apply to the time they actually start driving for

Lyft, pending background checks and other steps that contribute to applicant churn. We circumvent this delayed reward with a model that predicts the probability that a new driver applicant will start driving by a given time. While the details of this model are beyond the scope of this paper, we use its outputs in our system to quantify the expected number of driver acquisitions from an ad campaign.

1.2 Related Work

Budget allocation across different ad campaigns has been well-studied in the case where payouts, or returns on ad spend, are known. We apply similar budget allocation strategies as described in [7] and used by Criteo [6] and Netflix [11]. The key idea is that, assuming each ad campaign offers diminishing marginal returns, an advertiser's next unit of spend should always be spent on the campaign with the highest marginal payout. While this optimization deals with known payouts, the novelty of our system is its handling of risk and uncertainty.

Our approach to dealing with uncertainty in the payouts is a modified contextual bandit algorithm. Multi-Armed Bandit approaches have been used extensively throughout the adtech industry for manipulating creative [1, 5] and bidding/budgeting [8]. Our method relies on Thompson Sampling for managing the explore and exploit trade-off for ad campaigns with consistent performance history and those with greater uncertainty [2, 12]. It shares similarities with Contextual Bandits [3, 10] by supplying relevant contextual features to predict payouts and select the appropriate action. However our system differs from previous contextual bandit algorithms because we have a profitability/budget constraint. Our method is agnostic to different models for predicting payouts: any black-box regression model will work. Our method also extends these bandit approaches to ensure stable payout extrapolation far from the observed action-space of previously allocated budgets.

2 PROBLEM FORMULATION

2.1 Basic budget allocation

First we describe the basic budget allocation problem and its solution when the payouts are known and well-behaved. We follow the logic of previous work on advertising budget optimization [6, 7, 11], with minor modifications.

Consider a firm with a set of N investment options (or ad campaigns) operating concurrently. Each day, for a given ad campaign i, an allocated spend x_i yields payout $y_i = f_i(x_i) \geq 0$. We will refer to f_i as the payout curve or payout function for ad campaign i. Our system must prescribe target budgets $x_i \geq 0$ for all i on each day. Our goal is to maximize total payout under the total daily budget constraint B:

$$\begin{aligned} \underset{x \in \mathbb{R}}{\text{maximize}} \quad & \sum_{i=1}^{N} f_i(x_i) \\ \text{subject to} \quad & \sum_{i=1}^{N} x_i \leq B \\ & x_i \geq 0, \qquad i \in \{1, ..., N\} \end{aligned}$$

In the case of driver acquisition, the payout $y_i = f_i(x_i)$ is the number of new drivers acquired. If the typical value of a new driver is known to be C, we may use an alternative profitability constraint – we should never pay more than C for a new customer. This reformulation effectively supports infinite budget, provided acquisition remains profitable in expectation. The Cost Per Incremental Acquisition (CPIA) for an ad campaign is the inverse derivative of the payout function, $[df_i(x_i)/dx_i]^{-1}$. With this alternative constraint, our goal becomes

$$\begin{aligned} \underset{x \in \mathbb{R}}{\text{maximize}} \quad & \sum_{i=1}^{N} f_i(x_i) \\ \text{subject to} \quad & \left(\frac{df_i(x_i)}{dx}\right)^{-1} \leq C, \qquad i \in \{1, ..., N\} \\ & x_i \geq 0, \qquad i \in \{1, ..., N\} \end{aligned}$$

We anticipate acquisition to be profitable when the forecast customer value C exceeds the CPIA. At Lyft, C is provided externally based on regional forecasts for supply, demand, and driver utilization. The generation of these profitability targets is beyond the scope of this paper.

In this formulation, we assert the following desirable properties for forecast daily payout function f_i with respect to $x_i \geq 0$:

(1) f_i is differentiable (to compute CPIA)
(2) f_i is monotonically increasing (more spend always yields more drivers)
(3) f_i has sublinear growth (diminishing marginal returns)

In practice, we often observe a successful ad campaign has reached the entire available audience and no increase in budget can yield additional drivers, hence the diminishing marginal returns assertion. Furthermore, this property ensures campaigns with extraordinary returns have a limited budget growth rate day-over-day.

We also assume that the payout of one ad is independent of the payouts from other ads (no cannibalization). Under these assumptions, the optimal budget allocation is obtained by setting the budgets x_i such that the CPIA values for all ads are equal. Using the profitability constraint, we simply set x_i so that the CPIA equals the target customer value, $[df_i(x_i)/dx_i]^{-1} = C$, for all ad campaigns.

If the payout functions f_i are known, budget allocation is thus a solved problem. In reality the payouts are uncertain, so the problem reduces to generating forecasts $\hat{f}_i$ for each ad i from the available historical data. In the next subsection we describe challenges of producing such forecasts.

2.2 Challenges arising from payout uncertainty

Since the payouts are generally unknown, we develop a method to estimate the distribution of payout functions f_i and deal appropriately with their uncertainties. We highlight three challenges to estimating the payouts:

(1) Changing context: An ad may compete in an ad marketplace day after day, but its payout function may change over time (seasonality, ad auction competitor behavior, or the available audience and their preferences could change)
(2) Extrapolation: An effective model must predict payouts for levels of investment outside the range of past budgets.
(3) Cold start: We require payout predictions for new ads with little or no historical data.

2.3 Payout prediction as a contextual bandit

In order to deal appropriately with uncertainties in the payouts, we frame the problem as a contextual bandit. Our task is to set new budgets for all ads at each round in a sequence of rounds, with the goal of maximizing the total payout over the long-term. Each round we set a budget for each ad, then measure payout from each ad (in our case, the number of new drivers).

Our solution uses context features with supervised learning to predict the distribution of payouts of each ad campaign. It accomplishes exploration of new budgets by considering uncertainty in the payout distributions and sampling from the distributions. This cycle of predicting payouts, sampling from the prediction uncertainties, then allocating budgets repeats each round. Over time the system explores new budgets and collects data that make its predictions increasingly precise. In the following section we provide details of this method.

3 METHODS

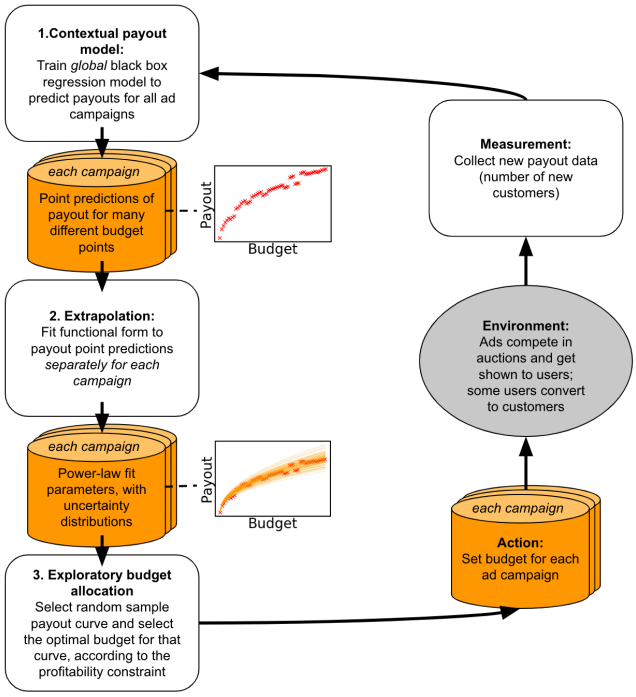

Figure 1: A schematic diagram of our system.

3.1 Contextual payout modeling

Our solution to predict payouts is to fit a global performance model on the entirety of the available advertisement performance history, including deprecated ad campaigns. The model predicts payouts (acquired acquisition) using advertising features (including ad copy, sign-on bonus), audience features (including region, target audience, day of week), and the daily expenditure (including actual advertising costs and expected bonus costs). Since the spend x_i is a feature of the model, we can form a predicted payout function $\hat{y}_i = \hat{f}_i(x_i)$, for each ad campaign, from the regression model results. Using a global regression model, rather than training separate models for each ad campaign, enables information-sharing among similar campaigns. For example, an ad creative that was successful in New York is also likely to succeed in San Francisco.

A simple non-linear model, such as Random Forest, has excellent performance *near-sample*, when the budget does not vary significantly from recent observations. However most supervised approaches cannot reliably extrapolate; in the extreme, tree-based model predictions are clipped by both the minimum and maximum observed target values. Accordingly these predictions cannot be used to forecast acquisition when target C is suddenly changed, so we require a separate extrapolation step.

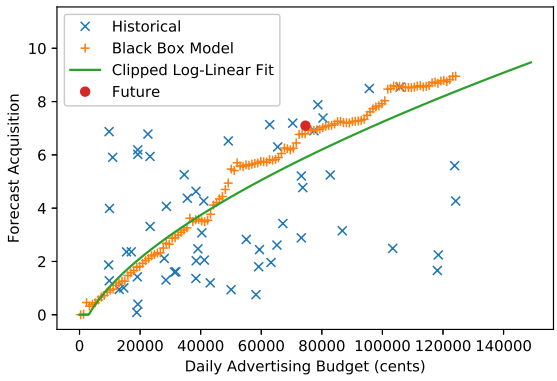

Figure 2: Payout curve extrapolation data is augmented by point predictions from a black box model.

3.2 Bayesian extrapolation

To forecast customer acquisition *far-from-sample*, where proposed budgets are far from the observed historical budgets, we fit a simple functional form to the payout function for each ad campaign using linear regression. The regression employs an augmented data set combining observed history for the campaign and a series of predicted returns generated by the contextual payout model. For data augmentation, the contextual model is fed a snapshot of the latest available context features for the campaign and a series of linearly-spaced proposal budgets chosen near the observed history, where the model interpolates reliably. Each campaign-specific linear regression provides a differentiable curve that is monotonically increasing and has diminishing returns (Section 2.1) using a power-law form $y = w_1 x^{w_2}$ (equivalently $\log(y) = w_1 + w_2 \log(x)$) [1] with w_1 non-negative and w_2 bounded $0 < w_2 < 1$. Figure 2 shows example results of fitting augmented data with this power-law model.

[1] In production, we fit $\log(y+1) = w_1 + w_2 \log(x+1)$ to support zero values, but refer to the simpler form above for brevity without significant algorithmic changes.

However, without uncertainty measures, this curve is insufficient for an exploration policy. We instead rely on Bayesian Linear Regression to provide covariance estimates for Gaussian distributed weights.

From design matrix $\boldsymbol{X}$ (containing historical budgets), historical acquisitions $\boldsymbol{y}$, prior inverse-gamma hyperparameters (a_0, b_0), prior precision matrix $\boldsymbol{\Lambda}_0$, and prior mean weights $\boldsymbol{\mu}_0$, we obtain posterior mean weights $\boldsymbol{\mu}$ and posterior precision matrix $\boldsymbol{\Lambda}$:

$$\boldsymbol{\mu} = (\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{\Lambda}_0)^{-1}(\boldsymbol{\Lambda}_0\boldsymbol{\mu}_0 + \boldsymbol{X}^T\boldsymbol{y})$$
$$\boldsymbol{\Lambda} = \boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{\Lambda}_0$$

In this framing, the vector $\boldsymbol{\mu} = (w_1, w_2)$ gives an estimate of the shape and normalization of the power-law fit. Covariance can be evaluated from normal-inverse gamma parameters a, b as follows [14],

$$\text{Cov} = \frac{b}{a-1}\Lambda^{-1}$$
$$a = a_0 + |\boldsymbol{y}|/2$$
$$b = b_0 + \frac{1}{2}(\boldsymbol{y}^T\boldsymbol{y} + \boldsymbol{\mu}_0^T\boldsymbol{\Lambda}_0\boldsymbol{\mu}_0 + \boldsymbol{\mu}^T\boldsymbol{\Lambda}\boldsymbol{\mu})$$

Thus we obtain an estimate of the power-law payout curve distribution, with an uncertainty measure provided by the parameter covariance. This estimate is generated independently for each ad campaign. The next step is Thompson Sampling from the payout curve distribution, which controls exploration of budgets.

3.3 Exploratory budget allocation

From parameter distributions $(w_1, w_2) \sim \mathcal{N}(\boldsymbol{\mu}, \text{Cov})$, we sample curves of the form $f(x) = w_1 x^{w_2}$ for Thompson Sampling independently for each campaign. Figure 3 shows an example set of curves sampled from the posterior distribution of the Bayesian Linear Regression model. Using Thompson Sampling, we randomly choose one curve, f_i^*, from the sample curves. Using f_i^* and the provided CPIA target C, we evaluate a target budget $\hat{x}_i$ where $[df_i^*(\hat{x}_i)/dx_i]^{-1} = C$ for each ad campaign i daily. Figure 4 shows how the allocated budget increases with CPIA target and varies among sampled curves. By randomly exploring curves that are more optimistic or pessimistic than the best-fit, the system allocates larger or smaller budgets, respectively. The system exhibits greater budget exploration if an ad campaign has substantial performance variance by sampling curves from broader parameter distributions. Over time, the system accumulates a larger and more diverse volume of data than that of a pure-exploitation model, improving forecast accuracy in the long term.

4 RESULTS

4.1 Baseline

Our previous model, like the new system described in Section 3, used a functional form for the payout curves as well as Thompson Sampling. It did not, however, incorporate a contextual regression model. Instead, the model would fit an independent curve to the historical payouts data for each ad campaign. Conceptually, the model used a hierarchical model that required Markov Chain Monte Carlo (**MCMC**) sampling.

Roughly, **MCMC** is a method for estimating the best-fit distribution of the parameters in a model by pseudo-randomly exploring

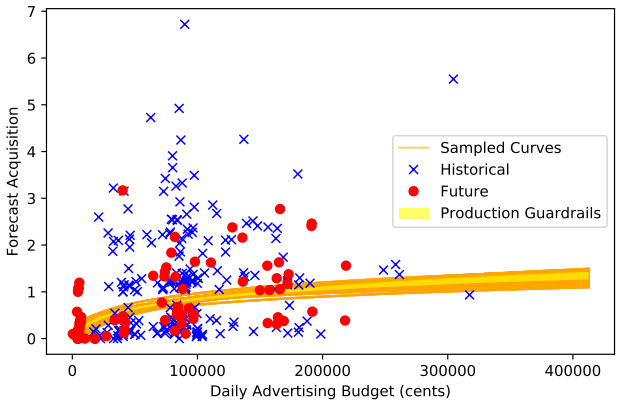

Figure 3: Payout curve distribution variance facilitates exploration with Thompson Sampling.

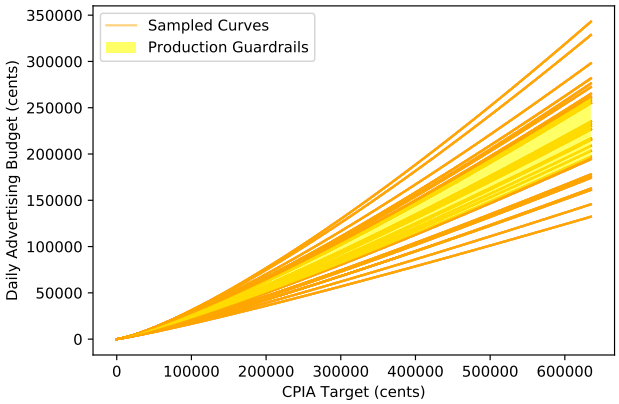

Figure 4: Budget curve distribution grows exponentially as a function of CPIA target.

points in the parameter space and accepting or rejecting them based on their consistency with observations. The baseline implementation applied the same form, $y = w_1 x^{w_2} + \epsilon$, assuming Gaussian error, $\epsilon \sim \mathcal{N}(\mu_\epsilon, \sigma_\epsilon^2)$. Weights are sampled $w_1 \sim \Gamma(\alpha_1, \beta_1)$ and $w_2 \sim \text{Beta}(\alpha_2, \beta_2)$, respectively.

4.2 Offline Evaluation

We compared the accuracy of payout predictions from several different regression models using historical data. We tested the previous implementation using **MCMC**, Least Squares Regression (**LSR**), Bayesian Linear Regression (**BLR**), Random Forest (**RF**), RF-augmented Least Squares Regression (**RF_LSR**), RF-augmented Bayesian Linear Regression (**RF_BLR**), and LightGBM (**LGBM**) [9]. We used training data with observations from January 1, 2019 - April 13, 2019 and testing on data spanning April 14 - 27, 2019.

RF has the lowest error in Table 1, however the model is not differentiable and cannot be applied directly to our maximization

Table 1: Offline Error (all advertisements)

Approach	Bias	MAE	MSE	Curve	Uncertainty
MCMC	0.1575	0.2144	0.5881	✓	✓
LSR	-0.6384	0.1095	0.4242	✓	✗
BLR	-0.1029	0.1565	0.4024	✓	✓
RF	**-0.0340**	**0.0788**	**0.0656**	✗	✗
RF_LSR	**-0.0364**	**0.0894**	**0.0871**	✓	✗
RF_BLR	**-0.0349**	**0.0937**	**0.0946**	✓	✓
LGBM	-0.0764	0.1894	0.3463	✗	✗

Table 2: Offline Error (cold start, less than 7 day data)

Approach	Bias	MAE	MSE	Curve	Uncertainty
MCMC	0.0842	0.123	0.0372	✓	✓
LSR	0.0364	0.1118	2.1390	✓	✗
BLR	**-0.0045**	**0.0586**	**0.0290**	✓	✓
RF	0.0179	0.0714	0.0421	✗	✗
RF_LSR	**0.0087**	**0.0668**	**0.0264**	✓	✗
RF_BLR	**0.0081**	**0.0682**	**0.0257**	✓	✓
LGBM	0.0366	0.0987	**0.0275**	✗	✗

Table 3: Online Error (all advertisements)

Approach	MAE	MSE
MCMC	0.1320 ± 0.0207	0.0716 ± 0.0467
RF_BLR	**0.0732 ± 0.0122**	**0.0280 ± 0.0132**

strategy. This limitation also holds for **LGBM**. And while the combined **RF_LSR** approach yields the best curve-fit, neither it nor **LSR** provides uncertainty measures for exploration in Thompson Sampling.

We also examined the subgroup of campaigns with less than 7 days of training data to evaluate cold-start in Table 2. New campaigns are typically allocated more conservative budgets from priors and initially generate fewer customer acquisitions until demonstrating cost-efficient returns. This presents as lower absolute values in the unscaled error metrics shown in Table 2 than in Table 1. Regardless, **RF_BLR** yields the lowest Mean Squared Error (MSE) among strategies meeting our maximization strategy criteria and was deployed for online experimentation.

4.3 Online Experiments

We measured our system using **RF_BLR** in a region-split A/B test over 7 major cities. We launched **RF_BLR** in 4 cities for 3 weeks while measuring 3 additional cities using the existing **MCMC** approach as control. After, we swapped control and treatment cities and continued experimentation for 3 additional weeks to measure changes within each city. During the control/treatment swap, we delayed measurement 1 week to mitigate experimental interference due to ad viewers who had been exposed to the previous strategy. For the duration of the experiment, we ensured the CPIA targets for these regions were unchanged.

Table 4: Experiment Results

City	CPA % Improvement	Std Error
Boston	37.08	24.02
Chicago	32.56	57.23
Denver	-0.01 (worse)	11.95
Philadelphia	15.57	20.88
Pittsburgh	79.03	10.73
San Francisco	3.09	9.93
Washington, DC	9.92	23.50

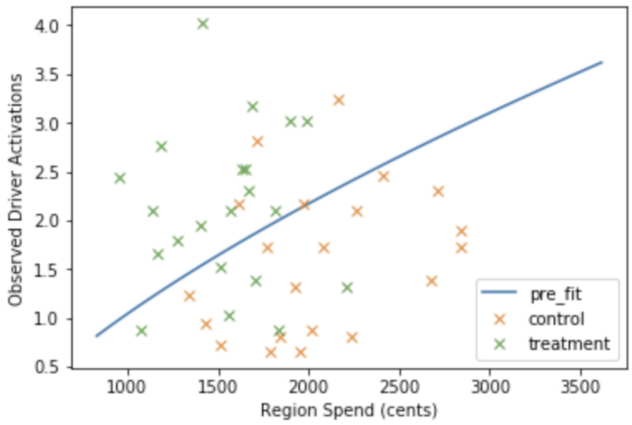

Figure 5: Daily driver acquisition using RF_BLR (treatment) yields more drivers at lower expense than the baseline approach using MCMC (control). The pre-fit curve is fit to measured returns prior to experiment launch.

In the online error evaluation, both systems are only responsible for prediction accuracy on a 1-day horizon (instead of up to 2-weeks in the offline evaluation). This presents as lower absolute error in Table 3 than in Table 1. Regardless, our system dominated the baseline approach across error metrics.

Because it is difficult to measure CPIA or Marginal Return on Investment directly [6], we instead measure the mean Cost Per Acquisition (CPA). From pre-experiment measurements, we use Ordinary Least Squares to fit a curve p, providing an estimated baseline performance function defined for all budget values x, including for unobserved budgets. We consider the daily experimental spend x and driver acquisition payout y as a ratio $y/p(x)$. We then evaluate the improvement of treatment over control as a ratio. Figure 5 illustrates this measurement for one region. Controlling for location in Table 4, we measure an overall CPA improvement of (21.8 ± 10.3)% between treatment and control. Similarly, we measure a (21.5 ± 13.1)% CPA improvement while controlling for time. These improvements correspond to tens of millions of dollars in savings for 2019. Shortly after the conclusion of this experiment, **RF_BLR** was deployed to all regions.

5 DEPLOYMENT

The system is scheduled to retrain the global performance model and allocate budgets nightly using Apache Airflow without human oversight. Although larger model architectures are available than Random Forest, this simple model is trusted to execute without additional hyperparameter tuning. However upstream data pipelines have a non-trivial failure rate. For robustness, we have implemented simple checks in the distributions of our inputs to filter potential errors, preferring stale data to erroneous data. Because Thompson Sampling yields randomly sampled budget allocations, the system continues to explore even when the latest data is not available. Furthermore, production exploration is restricted by guardrails, filtering both the lower and upper 25% of sampled curve variation within each advertising campaign.

In our framing, each payout curve is estimated $\hat{f}(x) = \hat{w}_1 x^{\hat{w}_2}$ and satisfies property (3) sublinear growth if $0 < \hat{w}_2 < 1$. For sparse or high-variance data, it is plausible that Bayesian Linear Regression fit a parameter distribution $\hat{w}_2 \sim \mathcal{N}(\mu_{w_2}, \sigma^2_{w_2})$ that can yield samples outside of this range. We resolve this by iteratively increasing the prior precision parameter Λ_{w_2} using values in $[0, 1, 4, 16, 64, 256]$, increasing the strength of prior mean weight $\boldsymbol{\mu}_0$ until the lower-confidence and upper-confidence bounds of $\hat{w}_2$ are both between $(0, 1)$. We then reject samples that do not satisfy property (3) sublinear growth.

Before manipulating real budgets, we conducted extensive hyperparameter optimization using multi-dimensional random searches, both offline and online off-policy evaluation. We searched scikit-learn Random Forest hyperparameters n_estimator $\in [10, 512]$ and max_features $\in \{\text{auto}, \log2, 0.1, 0.25, 0.5\}$ and Bayesian Linear Regression hyperparameters for the number of augmented data points $\in [128, 512]$ and prior precision $\Lambda_{w_2} \in [0, 49]$. The number of augmented data points is an especially sensitive hyperparameter because it correlates with the posterior precision of the parameter distributions generated by Bayesian Linear Regression.

5.1 Computational advantages of our system

In addition to CPA improvements, deploying our system reduced the infrastructure necessary for allocating budgets from 15 to 1 AWS r5a.8xlarge instances. It also improved reliability by eliminating the possibility of chain failure in **MCMC** due to poorly fitting priors and failure to converge model parameters within the allotted time. Deploying to **RF_BLR** increased the Mean Time Between Failures from 2.04 to 21.0 days in the 100 most recent execution attempts.

6 DISCUSSION AND CONCLUSION

In this paper, we present a framework for supporting budget exploration from black box machine learning models. Our system attempts to maximize total driver acquisition by allocating campaign budgets such that the mean cost of acquisition is profitable based on forecast driver supply and rider demand. Using Bayesian Linear Regression, our system transforms point predictions into payout curve distributions. Using Thompson Sampling for exploration, our system obtains diverse observations for future exploitation. In experiment, we measure a $(22 \pm 10)\%$ improvement in the mean Cost Per Acquisition over the existing Markov Chain Monte Carlo method while controlling for location and report similar results while controlling for time.

This work presents a proof-of-concept for other Reinforcement Learning systems at Lyft, including a dedicated Bandit Platform for orchestrating adaptive field experiments, similar to Facebook's Ax [4]. We have deployed the Platform to control in-application banner copy and measure a 40% improvement to conversion rate. We have running experiments using Contextual Bandits to rank banner campaigns based on expected Click-Through-Rate. We are also researching and developing an approach to dispatching drivers using online event-based Temporal Difference updates [15].

ACKNOWLEDGMENTS

The authors would like to thank Jared Bauman, Dongwei Cao, William Borges, Carolyn Conway, Antonio Luna, Jack van Ryswyck, Xing Xing, Alejandro Veen, Marisa Wong, Suma Snehalatha, Linda Dreyer, Michael Yoshizawa, Michael Lum, Diana Nedkova, Stefan Zier, Brittany Branscomb, Carolyn Crimi, Stephanie Gutierrez, Lei Tang, Siddharth Patil, Elizabeth Stone, Don Dini, Michael Rotkowitz, Ilan Lobel, Patrick McGrath, Ajay Sampat, Erik Vandekieft, and Gautam Kedia for their help on this project and paper.

REFERENCES

[1] Deepak Agarwal, Bee chung Chen, Pradheep Elango, Nitin Motgi, Seung taek Park, Raghu Ramakrishnan, Scott Roy, and Joe Zachariah. 2009. Online Models for Content Optimization. In *Advances in Neural Information Processing Systems 21*. 17–24.

[2] Shipra Agrawal and Navin Goyal. 2011. Analysis of Thompson Sampling for the multi-armed bandit problem. *CoRR* (2011).

[3] Peter Auer. 2002. Using confidence bounds for exploitationexploration trade-offs. *Journal of Machine Learning Research* 3 (2002), 397–422.

[4] Eytan Bakshy, Lili Dworkin, Brian Karrer, Konstantin Kashin, Benjamin Letham, Ashwin Murthy, and Sonia Singh. 2018. AE : A domain-agnostic platform for adaptive experimentation.

[5] Deepayan Chakrabarti, Ravi Kumar, Filip Radlinski, and Eli Upfal. 2009. Mortal Multi-Armed Bandits. *Advances in Neural Information Processing Systems 21* (2009), 273–280.

[6] Vincenzo D'Elia. 2019. On the causality of advertising. http://papers.adkdd.org/2019/invited-talks/slides-adkdd19-delia-causality.pdf [Online; accessed 10-December-2019].

[7] Paul W. Farris, Dominique M. Hanssens, James D. Lenskold, and David J. Reibstein. 2015. Marketing return on investment: Seeking clarity for concept and measurement. *Applied Marketing Analytics* 1 (apr 2015), 267–282.

[8] Junqi Jin, Chengru Song, Han Li, Kun Gai, Jun Wang, and Weinan Zhang. 2018. Real-Time Bidding with Multi-Agent Reinforcement Learning in Display Advertising. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management - CIKM '18* (2018).

[9] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.). 3146–3154.

[10] John Langford and Tong Zhang. 2008. The Epoch-Greedy Algorithm for Contextual Multi-armed Bandits. In *Advances in Neural Information Processing Systems 20*. 817–824.

[11] Randall A. Lewis and Jeffrey Wong. 2018. Incrementality Bidding Attribution.

[12] Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. 2010. A contextual-bandit approach to personalized news article recommendation. *Proceedings of the 19th international conference on World wide web - WWW '10* (2010).

[13] William R. Thompson. 1933. On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples. *Biometrika* 25, 3/4 (1933), 285–294.

[14] Gero Walter and Thomas Augustin. 2010. *Bayesian Linear Regression — Different Conjugate Models and Their (In)Sensitivity to Prior-Data Conflict.* 59–78.

[15] Zhe Xu, Zhixin Li, Qingwen Guan, Dingshui Zhang, Qiang Li, Junxiao Nan, Chunyang Liu, Wei Bian, and Jieping Ye. 2018. 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*. 905–913.