Boosted Decision Tree Regression Adjustment for Variance Reduction in Online Controlled Experiments

Alexey Poyarkov
Yandex, Moscow, Russia
alexeypoyarkov@gmail.com

Alexey Drutsa
Yandex, Moscow, Russia
adrutsa@yandex.ru

Andrey Khalyavin
Yandex, Moscow, Russia
halyavin@yandex-team.ru

Gleb Gusev
Yandex, Moscow, Russia
gleb57@yandex-team.ru

Pavel Serdyukov
Yandex, Moscow, Russia
pavser@yandex-team.ru

ABSTRACT

Nowadays, the development of most leading web services is controlled by online experiments that qualify and quantify the steady stream of their updates achieving more than a thousand concurrent experiments per day. Despite the increasing need for running more experiments, these services are limited in their user traffic. This situation leads to the problem of finding a new or improving existing key performance metric with a higher sensitivity and lower variance. We focus on the problem of variance reduction for engagement metrics of user loyalty that are widely used in A/B testing of web services. We develop a general framework that is based on evaluation of the mean difference between the actual and the approximated values of the key performance metric (instead of the mean of this metric). On the one hand, it allows us to incorporate the state-of-the-art techniques widely used in randomized experiments of clinical and social research, but limitedly used in online evaluation. On the other hand, we propose a new class of methods based on advanced machine learning algorithms, including ensembles of decision trees, that, to the best of our knowledge, have not been applied earlier to the problem of variance reduction. We validate the variance reduction approaches on a very large set of real large-scale A/B experiments run at Yandex for different engagement metrics of user loyalty. Our best approach demonstrates 63% average variance reduction (which is equivalent to 63% saved user traffic) and detects the treatment effect in 2 times more A/B experiments.

Keywords: A/B test; variance reduction; prediction

1. INTRODUCTION

Modern Internet companies improve their web services by means of data-driven decisions that are based on *online controlled experiments* also known as *A/B tests* [21]. The scale of use of this state-of-the-art technique, in particular, in search engines is impressive: Bing reported on more than 200 run experiments per day in 2013 (this number grew exponentially over the years [20]), and Google conducted more than 1000 experiments in any day in 2015 [16]. Since the user traffic is limited for a web service, it is vital to effectively use it for maintaining the upward trend of A/B testing.

An A/B test compares two variants of a service at a time, usually its current version (control) and a new one (treatment), by exposing them to two groups of users. The aim of controlled experiments is to detect the causal effect of service updates on its performance relying on an *Overall Evaluation Criterion (OEC)* [23], a user behavior metric (e.g., clicks-per-user, sessions-per-user, etc.) that is assumed to correlate with the quality of the service. The ability of an A/B test to detect the statistically significant difference when the treatment effect exists is referred to as the *sensitivity* [23]. Sensitivity of a particular A/B test could be increased either by a larger sample of participated users, or by a more powerful (more sensitive) OEC. Since user traffic is limited, OEC's sensitivity becomes a crucial aspect that affects the number of experiments with detected treatment effect [23, 21]. That is why a wide set of studies [23, 7, 30, 6, 21, 5, 10, 8, 18, 29, 11] addressed the sensitivity and its improvement.

In this context, the engagement metrics of user loyalty are of greatest interest, since, on the one hand, they are predictive of long-term goals of Internet companies [19, 20, 21] and often considered to be most appropriate for online evaluation (e.g., sessions-per-user [30] is accepted as the "North-star" for online controlled evaluation in major search engine companies like Bing [20, 21]). On the other hand, they are very insensitive to changes of a service [21] what results in utilization of a very large amount of user traffic to achieve a desired level of sensitivity to such small changes (usually, experiments span weeks and cover hundreds of thousands users). Previous work on sensitivity improvement for the loyalty metrics has been limited either to alternative key metrics (periodicity [9, 8] and transformation [11]), evaluation statistics [11], and statistical tests [11], or to a virtual increase of the duration of an experiment by peaking into the future through prediction of the key metric [10].

Since the variance of a key metric reflects its noisiness, a promising way to improve the sensitivity is to decrease the variance [7]. First, the existing studies are limited to only one form of the control variate technique and to only one covariate in the context of large-scale online A/B tests. Second, the empirical validation of the proposed approaches is scant: only a couple of A/B tests is considered and the achieved variance reduction rate is reported only approxi-

KDD '16, August 13–17, 2016, San Francisco, CA, USA.

DOI: http://dx.doi.org/10.1145/2939672.2939688

mately ("up to 40–50%"). Finally, the approaches were not applied to engagement metrics of user loyalty, while, as mentioned above, they are less sensitive than the ones considered in the study, what may have had an effect on the reported variance reduction rate.

Thus, in our study, we address the problem of variance reduction (VR) for user engagement metrics and develop a general framework that allows us to incorporate both the existing state-of-the-art approaches to reduce the variance and some novel ones based on advanced machine learning techniques. We are motivated by the following intuition. The expected value of the key metric for a given user consists of two components: (1) the expected value for this user irrespectively the treatment assignment and (2) the expected bias of the value due to the treatment assignment (i.e., the treatment effect for this user). Since the expectations of the first component does not depend on the treatment assignment, this component does not contribute to the actual average treatment effect, but may increase the variance of its estimation. If we knew the value of the first component, we would subtract it from the key metric and obtain a new metric with decreased variance. However, since we cannot evaluate the first component exactly, we propose to predict it based on the attributes of the user that are independent of the treatment exposure. Therefore, we propose to utilize, as an OEC, instead of the average value of a key metric, its average deviation from its predicted value. We show that the variance of the modified metric is proportional to the mean square error of the predictor. In this way, the problem of VR is reduced to the problem of finding the best predictor for the key metric that is not aware of the treatment exposure.

The framework of our approach covers both the variance reduction technique [7] earlier applied to online evaluation and the ones that are well known in randomized experiments in medicine, social sciences, etc. [14, 28], but, to the best of our knowledge, were never applied in the case of large-scale online A/B tests. At the same time, in our general approach, we apply gradient boosted decision trees (that, as far as we know, were never used to reduce the variance) and achieve a significantly greater variance reduction than the methods of the previous works. In this paper, we conduct our experimental analysis on 161 large-scale A/B experiments run on real users of Yandex (www.yandex.com), one of the most popular global search engines, with duration from one to several weeks and user samples from $5 \cdot 10^5$ to $3 \cdot 10^7$ users. This should make the results of our study more valuable for practical use in modern Internet companies.

To sum up, our study focuses on the problem, which is recognized as fundamental for the *present and emerging Internet companies' needs*: to conduct *more* online controlled experiments per day. Specifically, the major contributions of our work include:

- A new class of variance reduction methods based on advanced machine learning, including ensembles of decision trees, that, to the best of our knowledge, were never used to reduce variance.
- Validation of our variance reduction approach on 161 real online experiments run at Yandex for different engagement metrics of user loyalty, first, demonstrating statistically significant 5.1% improvement of average variance reduction over the best state-of-the-art VR technique (which is equivalent to overall 63% saved user traffic) and, second, detecting the treatment effect in +20% more A/B tests than the best state-of-the-art VR technique.

The rest of the paper is organized as follows. In Sec. 2, the related work on controlled experiments and user engagement is discussed. In Sec. 3, we remind the key points of A/B testing and introduce the general framework of our variance reduction approach. The user engagement metrics and the data (in particular, our 161 A/B experiments) used to validate our approach are described in Sec. 4. In Sec. 5, the details of our prediction task are discussed and our prediction models are evaluated with respect to different settings. In Sec. 6, we validate our approach with respect to variance reduction and sensitivity improvement of key user engagement metrics. In Sec. 7, the study's conclusions and our plans for the future work are presented.

2. RELATED WORK

Online controlled experiment studies. A/B testing methodology achieved a very high popularity in Internet industry over the past few years what is reflected in the recent explosion in the number of published studies on this topic. Early studies [22, 23] were devoted to the theoretical aspects of the methodology. Subsequent work included studies of various aspects of the application of A/B testing in Internet companies: evaluation of changes in various components of web services (e.g., the user interface and ranking algorithms [30, 9, 29]); large-scale experimental infrastructure [31, 20] and optimal scheduling of the experimentation pipeline [17]; different parameters of user interaction with a web service (speed [21], absence [12], abandonment [21], periodicity [9, 8], and engagement [9, 10, 8, 11]); effects of long-term user learning [16]. Many "rules of thumb", pitfalls, and puzzling outcomes of online controlled experimentation were summarized in several studies [4, 19, 21] devoted to the trustworthiness of A/B test results.

A substantial number of studies were devoted to the problem of improving sensitivity of online experiments and saving user traffic in them, what reflects the actual needs of a modern Internet company: to detect the treatment effect in more experiments while expending available resources optimally. Some approaches (to improve sensitivity or to optimize resources) are focused on the alterations of the user groups involved in an A/B experiment (e.g., expanding of user sample [23], elimination of users who were not affected by the service change in the treatment group [30, 5]), as well as changes of the experiment duration (either real increasing [23], or virtual one through the prediction of the future [10]). Some other studies address the problem through a search of more sensitive metrics and their transformations [21, 9, 8] or through the use of more appropriate statistical tests and overall acceptance criteria: statistical tests for two-stage A/B experiments [6], sequential testing for early stopping [18], the optimal distribution decomposition approach [29], and thorough comparison of different evaluation statistics and statistical tests [11]. Finally, this problem is addressed by the variance reduction techniques that are known from Statistics [28] and digital simulation [32]: the stratification and control covariates [7].

Overall, the studies [10] and [7] are the most relevant ones to our work in the context of online controlled experiments. Drutsa et al. in [10] utilized prediction of user engagement to improve sensitivity of an A/B experiment in another way than we do. They used user behaviour observed during the

experiment period to predict the value of an engagement metric of each individual user in a *future (post-experiment) period*, and, then, they used these values as a *more sensitive key metric*. While, in our study, we use user behaviour observed during the *pre-experiment period* to predict a key engagement metric of each individual user in the *experiment period*, and, then, we improve the sensitivity of the key metric by *subtracting the predicted values* from the actual values of the key metric. Deng et al. in [7] are the first who proposed to utilize pre-experiment data in order to reduce the variance of a key metric. However, their study was limited to the basic forms of control variates techniques that were based only on one feature and were validated via scant empirical analysis (over a pair of A/B tests). This technique is a particular case of our general framework (i.e., the linear model based on one covariate) and, therefore, is considered as a baseline in our study. Besides, we experiment with a very large and diverse set of 161 large-scale A/B tests based on actual interactions of hundreds of thousands of real users.

Randomized experiments in general. Actually, the idea proposed by Deng et al. [7] is not novel in the context of the statistical theory of randomized experiments (randomized control trials, etc.) and their widespread application in clinical researches and other research areas (e.g., the social sciences). Many variance reduction techniques, initially considered in digital simulation [32, 24], were also actively used in the randomized experiments [14, 28]. There are methods that do not change experiment randomization design (e.g., control variates) and methods that change it (blocking, pre-stratification, re-randomization, etc. [1, 27]). Control variates approach is based on a linear model of several covariates (also known as regression adjustment [13]) that approximates the key metric (either by the usual method of ordinary least squares [13, 25], or by a more complex one like Lasso [3]). This technique is a particular case of our general framework (where the learning model built on all available covariates is linear) and, therefore, is also considered as a baseline one in our study (as a straightforward extension of Deng et al.'s approach [7]). In our study, we also consider a technique of matching, which is often used in observational studies [14, 28] to reduce bias in the treatment effect estimation, but could be applied for variance reduction in randomized experiments as well.

Overall, in the context of randomized experiments, on the one hand, our work addresses and provides a verification of the state-of-the-art variance reduction technique on a very large set of large-scale online experiments with at least hundreds of thousands experimental units (in contrast to clinical and social studies). On the other hand, we apply the advanced machine learning method (gradient boosted decision trees[1]) that noticeably improves the state-of-the-art variance reductions, and, thus, it should be of interest from the perspective of randomized experiments in general.

3. FRAMEWORK

3.1 A/B testing background

Assume that we need to compare the performance of a new variant B (*the treatment*) of a web service and the current production variant A (*the control*) w.r.t. a *key metric* X, which quantifies user behavior[2]. Formally, one needs to estimate the *average treatment effect* (ATE) defined as

$$\mathrm{ATE}(X) = \mathrm{E}(X \mid B) - \mathrm{E}(X \mid A). \quad (1)$$

In an A/B test (a randomized experiment) [23, 19, 21, 14, 28], users (from a set $\mathcal{U}$ referred to as the *user traffic*), participated in the experiment, are randomly exposed (assigned) to one of the two variants of the service (i.e., $\mathcal{U} = \mathcal{U}_A \sqcup \mathcal{U}_B$). In order to estimate $\mathrm{ATE}(X)$, one estimates each term in Eq. (1) by the average values $\mu_V(X) = \mathrm{avg}_{\mathcal{U}_V} X, V \in \{A, B\}$ (given the observations of metric X for each user group $\mathcal{U}_V, V \in \{A, B\}$), where $\mathrm{avg}_\Omega X = \sum_{\omega \in \Omega} X(\omega)/|\Omega|$ is the *Overall Evaluation Criterion* (*OEC*, also known as the evaluation metric, the online service quality metric, etc. [23]). Their difference $\Delta(X) = \mu_B(X) - \mu_A(X)$ is used as an estimator of $\mathrm{ATE}(X)$ to quantify its sign and magnitude.

The absolute value $|\Delta(X)|$ of the estimator should be controlled by a statistical significance test that provides the probability (called *p-value* and also known [11] as the *achieved significance level*, ASL) to observe this value or larger under *the null hypothesis*, which assumes that the observed difference is caused by random fluctuations, and the variants are not actually different. If the p-value is lower than the threshold $p_{\mathrm{val}} \leq \alpha$ ($\alpha = 0.05$ is commonly used [23, 7, 21, 9, 10]), then the test rejects the null hypothesis, and the difference $\Delta(X)$ is accepted as statistically significant. The pair of an OEC and a statistical test is referred [11] to as an *Overall Acceptance Criterion* (OAC). In our study, we utilize the widely applicable *two-sample t-test*[3] (as in [7, 30, 6, 9, 5, 10]). This test is based on the *t-statistic*:

$$\Delta(X) / \sqrt{\sigma_A^2(X) \cdot |\mathcal{U}_A|^{-1} + \sigma_B^2(X) \cdot |\mathcal{U}_B|^{-1}}, \quad (2)$$

where $\sigma_V(X)$ is the standard deviation of the metric X over the users $\mathcal{U}_V, V = A, B$. The larger the absolute value of the t-statistic, the lower the p-value. The additional details of the A/B testing framework could be found in the survey and practical guide on online experiments [23] or in some books on randomized experiments in general like [14, 28]. A practical comparison of different key metrics, evaluation statistics, and statistical tests on a large set of online experiments could be found in [11].

3.2 Variance reduction via subtraction of prediction

In this subsection, we introduce our general framework of the studied variance reduction approach, while, in the next one, we show how it could incorporate different particular cases investigated previously in the literature. To begin, the definition (2) of t-statistic implies that the p-value may be reduced either by an increase of the sample size $|\mathcal{U}|$, or by a reduction of the sample variance $\sigma^2_{\mathcal{U}}(X)$. Hence, a

[1]Similar approaches were applied to reduce bias in the treatment effect estimation in observational studies [26], but, to the best of our knowledge, were not applied to VR problems.

[2]From here on in this paper, we consider "per user metrics" [4, 11], which are calculated for each individual user. This type of metrics (e.g., the number of sessions for a user) is a popular choice for web services [19]. However, there are also frequently used non-per user metrics [4] like presence-time-per-session [11], the annual revenue [21], etc.

[3]The key metric may not follow some assumptions underlying this test, such as the normality of the metric's distribution. However, for large user samples, like those used in our study, the statistical test is correctly applicable [11] for engagement per-user metrics considered in this study.

reduction of the variance by a factor $\varkappa$ allows us to reduce the sample size $|\mathcal{U}|$ by the same factor $\varkappa$ while preserving the sensitivity level achieved before the variance reduction. Hence, *the percent of reduced variance is equal to the percent of saved user traffic, while preserving the same conclusions made in the experiments.*

The motivation. Suppose that we are able to characterize a user u by a set of attributes $\mathbf{F}_u \in \mathbb{R}^n$ that are independent of the treatment assignment $V \in \{A, B\}$. We represent the value of the key metric X as $X = M_1(\mathbf{F}) + M_2(\mathbf{F}, V) + X'$, where $M_1(\mathbf{F}) = \mathrm{E}(X \mid \mathbf{F})$ is the expectation of the key metric X over users with the attributes $\mathbf{F}$, $M_2(\mathbf{F}, V) = \mathrm{E}(X \mid \mathbf{F}, V) - \mathrm{E}(X \mid \mathbf{F})$ is the expectation bias of the metric for users with a given assignment V, and $X' = X - \mathrm{E}(X \mid \mathbf{F}, V)$ is a noise, i.e., the unpredictable part of the key metric. Then the OEC's $\Delta(X)$ consists of three terms: $\mu_B(M_1(\mathbf{F})) - \mu_A(M_1(\mathbf{F}))$, $\mu_B(M_2(\mathbf{F}, B)) - \mu_A(M_2(\mathbf{F}, A))$, and $\mu_B(X') - \mu_A(X')$. The second term is an unbiased estimator of $\mathrm{ATE}(X)$, while the first and the third ones do not affect it, since their expectations are zero. They only make contribution to the variance of the estimate. If we could accurately measure the value of $M_1(\mathbf{F})$ and X' on the basis of the available data about users $(\mathbf{F}_u, V_u)$, we would have subtracted them from our key metric, and, thus, would improve our OEC. In fact, we cannot calculate the third term, but the first term can be *approximated*.

The approach. Let $\widetilde{X}$ be a predictor of the key metric X such that $\widetilde{X}$ does not depend on the treatment exposure. Then *we propose to utilize the difference $\hat{X} = X - \widetilde{X}$ between the actual metric value X and the predicted one $\widetilde{X}$ as a novel key metric in the OEC.* We reveal two properties of the proposed key metric $\hat{X}$, which make it more effective than the original metric X. First, we note that

$$\begin{aligned}\mathrm{ATE}(\hat{X}) &= \mathrm{E}(X - \widetilde{X} \mid B) - \mathrm{E}(X - \widetilde{X} \mid A)\\ &= \mathrm{E}(X \mid B) - \mathrm{E}(X \mid A) - (\mathrm{E}(\widetilde{X} \mid B) - \mathrm{E}(\widetilde{X} \mid A)) \qquad (3)\\ &= \mathrm{ATE}(X) - (\mathrm{E}(\widetilde{X}) - \mathrm{E}(\widetilde{X})) = \mathrm{ATE}(X),\end{aligned}$$

where we used the independence of the predictor $\widetilde{X}$ with respect to the treatment. Thus, the difference $\Delta(\hat{X})$ of the novel metric is an unbiased estimator of the treatment effect for the source metric X. Hence, the OEC based on $\hat{X} = X - \widetilde{X}$ could be used instead of the one based on X.

Second, we introduce the following assumptions that may be satisfied by a predictor: (A1) a predictor minimizes the *Root Mean Square Error*, *RMSE* (i.e., it is optimal) in a class of learning models, which is (A2) closed under addition of a constant and (A3) closed under scalar multiplications.

Then, on the one hand, the variance of the metric $\hat{X}$ is

$$\mathrm{Var}(\hat{X}) = \mathrm{E}\big((X - \widetilde{X})^2\big) - \big(\mathrm{E}(X - \widetilde{X})\big)^2, \qquad (4)$$

where the first term is, by definition, the square of the RMSE of the predictor $\widetilde{X}$ and the second term is equal to 0 in the case of an unbiased predictor $\widetilde{X}$ (i.e., if $\mathrm{E}(\widetilde{X}) = \mathrm{E}(X)$, what holds under the assumptions (A1,2)). In this case, the identity (4) transforms to

$$\mathrm{Var}(\hat{X}) = \mathrm{RMSE}^2(X, \widetilde{X}), \qquad (5)$$

which states that the variance of the novel metric $\hat{X} = X - \widetilde{X}$ is directly proportional to the loss of the predictor $\widetilde{X}$ (in terms of the MSE). In this way, the better the predictor of the key metric X, the lesser the variance of the modified metric $\hat{X}$, and thus the lesser the volume of the user sample $\mathcal{U}$ required to obtain a certain confidence level.

On the other hand, we know (see, e.g., [10]) that, if the predictor $\widetilde{X}$ is unbiased and satisfies the assumptions (A1,3), then the following identity holds: $\mathrm{Var}(X) = \mathrm{Var}(\widetilde{X}) + \mathrm{Var}(X - \widetilde{X})$, which implies

$$\mathrm{Var}(\hat{X}) = \mathrm{Var}(X) - \mathrm{Var}(\widetilde{X}). \qquad (6)$$

Intuitively, the subtracted term $\mathrm{Var}(\widetilde{X})$ corresponds to the variance of the term $M_1(\mathbf{F})$ in the motivation above. The last identity shows that the variance of the difference $\hat{X}$ is lower than the variance of the source metric X in the case when our predictor is not a constant and satisfies (A1,2,3). These conditions are satisfied by a wide range of prediction model classes including linear models and the state-of-the-art ensembles of decision trees [15].

To sum up, (a) Eq. (3) implies that the ATE for the refined metric $\hat{X}$ is equal to the ATE for the source metric X; (b) the direct relationship between the variance of the refined metric $\hat{X}$ and the quality of the unbiased predictor $\widetilde{X}$ is stated in Eq. (5); (c) Eq. (6) guarantees a variance reduction in the case of a non-degenerate predictor $\widetilde{X}$ satisfying the conditions (A1,2,3).

3.3 Details of prediction and special cases

A fundamental approach to obtain a predictor of some target quantity for a set of entities $\mathcal{O}$ is to construct a map M, which depends only on n entity's features $\mathbf{F}(o) = (F_1(o), ..., F_n(o)), o \in \mathcal{O}, n \in \mathbb{N}$. In our case, set $\mathcal{O} = \mathcal{U}$ consists of users, M is referred to as the prediction model, and the target is the key metric X. Thus, $\widetilde{X}(u) = M(\mathbf{F}(u))$. In our approach, it is crucial that the features $\mathbf{F}$ are independent of the treatment assignment. In Sec. 5, we narrowly discuss the targets, features, and models used in our study.

Training set. In order to obtain an optimal map M that has the best prediction quality (i.e., the RMSE in our case) among models from a class $\mathfrak{M}$, a machine learning technique is applied to a training set of examples. According to the usual practice, the prediction process is conducted in the following manner: (a) we train and retrieve an optimal model M based on some examples, whose target is known; (b) we apply this model to the entities whose target is currently unknown. Following this, in our study, we collect some dataset of user behavior observed earlier than the time when our A/B experiments are conducted and use an earlier starting time point when calculating both the feature values and the target to train the model. After that, we use this model to predict the key metric measured in our experiments (e.g., as it is done in [10]). We call such a model a *global predictor*.

However, our VR approach does not require a prediction model prior to a conducted experiment, since the calculation of the OEC (where the predictor $\widetilde{X}$ is used) is performed only after the values of the source metric X are known. At the same time, we know that user behavior significantly depends on a time period, where this behavior is considered [9]. Therefore, the relationship established between features $\mathbf{F}$ and the target X by a model M may differ over different periods. These observations motivate us to train the model on a dataset of examples from the time period, where the obtained predictor will be applied to reduce the variance

of the key metric. We expect that this will lead to an improvement of the prediction quality compared to the global predictor. In this way, we obtain a *local predictor*, which is trained individually for each experiment.

Moreover, we can use any subset of the experiment's user traffic (both from the treatment and the control groups) as the training set, since the model outputs $M(\mathbf{F})$ will satisfy the condition of independence of the treatment assignment as far as they depend only on features $\mathbf{F}$ that satisfy this condition themselves (*any map M is independent of users itself and, thus their treatment assignment*). For a large dataset like the one used in our experiments, we did not observe any significant overfitting to the training set and we also did not observe any decrease in the variance reduction rate when using the same dataset of users to train a predictor and to estimate ATE(X). We compare global and local predictors in our experimentation presented in Sec. 5.4 and 6.

Control variates. In a particular case, where we learn a linear predictor $M(F) = \theta F, \theta \in \mathbb{R}$ based on only one feature F, we obtain the approach considered in [7]. The best model is determined as the ordinary least square (OLS) solution with $\theta_0 = \mathrm{Cov}(F, X)/\mathrm{Var}(F)$ estimated either from the considered experiment's data, or from a period before the experiment. This variance reduction technique is known as *control variates*, and, to the best of our knowledge, [7, 5] are the only studies, where this technique is applied to large-scale online experiments (with only one covariate in both cases). However, this technique (also known as linear regression adjustment) is thoroughly studied in the theory of randomized experiments and widely used in offline studies (like clinical or social ones) for more than one covariate as well [13, 14, 25, 28, 3]. Namely, in terms of our general framework, this technique is based on the class of linear models (i.e., $M(\mathbf{F}) = a_0 + \sum_{j=1}^{n} a_j F_j, (a_j)_{j=0}^{k} \in \mathbb{R}^{n+1}$), where the best one is usually determined by the OLS solution [13, 14, 25, 28]. Thus, this state-of-the-art variance reduction technique is a particular case of our approach. To the best of our knowledge, the technique was never previously applied to large-scale online A/B testing studies based on several covariates. We apply it in our empirical study both to one (as in [7]) and to all available features (Sec. 5.4 and 6).

Machine learning. The crucial peculiarity of online experiments is large amounts of user data that need to be processed (usually at least hundreds of thousands experimental units) and complicated dependence of evaluated metrics on covariates, hence we believe that advanced methods of machine learning would work *especially effective* in this case. Therefore, in our paper, we *propose to apply them to the problem of variance reduction.* Namely, we find the optimal prediction model in the class of ensembles of decision trees by means of the state-of-the-art Friedman's gradient boosted decision tree (GBDT) method [15]. To the best of our knowledge, no existing study on variance reduction in randomized experiments (both online and offline) investigated such a machine learned model. The fact that linear regression adjustment [13, 14, 25, 28, 3] uses the same data for learning parameters and for measuring adjusted metric supports our idea to proceed in a similar manner, when we apply GBDTs.

Matching. This approach is based on the idea of finding a similar user (or a set of similar users) from the group $\mathcal{U}_A$ ($\mathcal{U}_B$) for each user from the other group $\mathcal{U}_B$ ($\mathcal{U}_A$ respectively). Matching is usually applied to reduce the bias of a

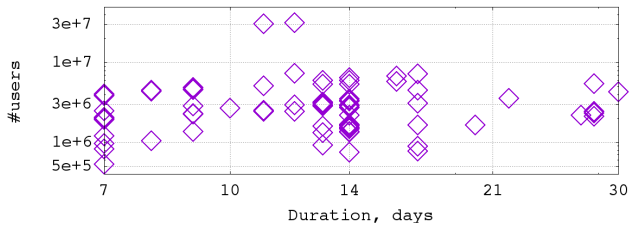

Figure 1: Joint distribution of 161 studied A/B tests w.r.t. their duration and the user traffic.

treatment effect estimation in the context of observational studies, when the randomized assignment is not possible [14, 28], but it also could be used in randomized experiments as a particular case of our general framework. In most matching techniques (e.g., k-nearest neighbours, kNN), the criteria of the similarity is based on a distance in the space of user features $F \in \mathbb{R}^n$. In terms of machine learning, for a user u, the value $X(u_m)$ of the matched user u_m (or a combination of the metric's values for a set of matched users $\{u_m\}$) could be considered simply as a approximated value of the metric for the user u (e.g., estimated by kNN) that could be observed if user u would be assigned to the other group. Hence, first, the machine learning methods used in matching could also be utilized in our approach, and, second, advanced machine learning methods could be applied in matching techniques. In the first case, machine learning methods usually used in matching (like kNN) [14, 28] are very computationally costly considering the sizes of our user samples (from $5 \cdot 10^5$ to $3 \cdot 10^7$, see Sec. 4) and, thus, are infeasible for our large-scale experiments. In the second case, one could apply the state-of-the-art GBDT method [15] to get the following *matching estimator* of the ATE(X)[4]:

$$\Delta_{match}(X) = \frac{|\mathcal{U}_B|}{|\mathcal{U}_A \cup \mathcal{U}_B|}\Delta(\hat{X}_A) + \frac{|\mathcal{U}_A|}{|\mathcal{U}_A \cup \mathcal{U}_B|}\Delta(\hat{X}_B), \quad (7)$$

where $\widetilde{X}_V$ is a predictor of the metric X trained on users $\mathcal{U}_V$, $V = A, B$ (via GBDT, in our study), $\hat{X}_A(u) := X(u) - \widetilde{X}_A(u)$ and $\hat{X}_B(u) := X(u) - \widetilde{X}_B(u)$.

4. STUDIED METRICS AND A/B TESTS

User engagement metrics. In this paper, we concentrate on the study of the loyalty aspect of user engagement [30, 12, 10], since, on the one hand, the metrics measuring this aspect are predictive of long-term goals of Internet companies [19, 20, 21] and are widely used in A/B testing practice [21, 10, 11]. On the other hand, these metrics are difficult to shift by a web service update [21]. Hence, even in the case when such metric X is able to catch the treatment effect during an A/B test, its ATE(X) is expected to be small in comparison with the variance of the metric. Therefore, catching such small effect with a desired statistical significance level will most probably require more resources (more users participating in the experiment over a longer period) than by means of such easily changeable activity-related metrics as, for instance, the number of clicks [21]. This fact was also observed in the recent comparisons [9, 10, 8, 11]: the activity metrics detect the treatment effect in up to 4 times more experiments than the loyalty ones.

[4]Due to space constraints, we omit details and refer a reader to [14, 28], where Eq. (7) is derived for kNN.

Following common practice [19, 12, 30, 10], we define a session as a sequence of user actions whose dwell times are less than 30 minutes. In this paper, we use browser cookie IDs to identify users as done in other studies on user engagement and online A/B testing [31, 12, 30, 9, 16]. We study two key metrics X for a user: the number of her sessions S (as in [30, 9, 10, 11]) and the absence time per session $ATpS$, which is measured as the total absence time (the duration of the whole time period, where the key metric is measured, minus the sum of the durations of all her sessions) divided by the number of sessions S (as in [10, 8, 29]). Due to space constraints, we mainly discuss and analyse the details of our approach for the state-of-the-art sessions-per-user OEC in Sec. 5 and 6. But the final empirical validation of the effectiveness of our approach is also reported for metrics S, $ATpS$, and for some of their modifications (see Sec. 6.3).

Our A/B experiments. In our paper, we consider 161 large-scale A/B experiments conducted on *the users* of Yandex with duration from 7 to 30 days lasted in 2013 and 2014 years. The user samples used in our A/B tests are all uniformly randomly selected, and the control and the treatment groups are approximately of the same size. The total number of users participated in each experiment varies from $5 \cdot 10^5$ to $3 \cdot 10^7$ users. The joint distribution of these 161 A/B tests with respect to their duration and the size of their sample of users is presented in Figure 1. Each experiment evaluates a change in one of the main components of the search engine (including the ranking algorithm, the user interface, the server efficiency, etc). Each of these changes is either an artificial deterioration of a search engine component[5] [20], or its update, which is evaluated before being shipped to production. Each experiment is verified against the absence of a carry-over effect [19] from the past, i.e., we explicitly check that there is no statistically significant difference in the considered OEC between the user groups in the 2-week period before the experiment.

5. PREDICTION

In this section, due to space constraints, only the number of sessions is considered as our *prediction target*.

5.1 Prediction data

The user behavior data are collected both from the period of an experiment (*the experiment period*) and from the 2-week period before the experiment (*the pre-experiment period*). The data from the experiment period are used to obtain the key metric, while both periods are used to obtain features utilized by a predictor of the key metric. Hence, our user engagement prediction problem has the following setting. One has user behavior data observed during two consecutive time periods. Then, one needs to estimate the value of a target engagement metric calculated over the second period for each *individual user* based only on his behavior (observed in both periods) which is not affected by the variant of the web service. Some investigations of the length of the pre-experiment period could be found in the context of the future user experiment prediction in [10] and in the context of variance reduction in [7].

In order to train our prediction models, we either utilize the user behavioral data from the experiments' periods (in this case, we get an individual local predictor for each experiment), or utilize the data collected far before these periods (in this case, we get one global predictor for experiments of the same duration), see Sec. 3.3. For the latter purpose, we additionally collected the behavioral data from 2013, by randomly selecting users and 3-week periods, in which the target is calculated over the last week and the two first weeks are used to calculate features (like for a pre-experiment period in an A/B test). As a result, we obtained a training set with $2.5 \cdot 10^5$ examples, which are then used to train a global predictor for 1-week A/B experiments (see Sec. 5.4 and 6.1).

[5]like the swap of the second and the fourth results in the ranked list returned by the current ranking [9, 29].

Table 1: Comparison of feature sets and models in terms of the average value of nRMSE over 161 A/B tests (relative improvement w.r.t. the first row).

Feature set \ Model	Linear Regression		Decision Trees	
Total	0.8823	(0%)	0.8972	(0%)
Total ∪ TS	0.8594	(−2.59%)	0.8531	(−4.91%)
Total ∪ TS ∪ CT	0.8479	(−3.9%)	0.8213	(−8.46%)
All	0.8418	(−4.59%)	**0.8203**	(−8.57%)

5.2 Features

In our prediction models we utilize the following features to predict the value of the target metric. Note that all of them are measured based on events that are not affected by the service version observed during the experiment.

The total feature. First, we consider the value of the key metric X calculated over the pre-experiment period as our main feature. On the one hand, this feature is reported in [10] as the most predictive one of the value of X in the future period (i.e., the period of an experiment in our case). On the other hand, such feature is known as an effective one in the control variate technique considered to reduce variance in [7]. We denote this feature by **Total**.

The time series. Second, we use the values of the key metric X calculated over each day of the pre-experiment period, obtaining a daily time series of length 14. Then, for each day, except the first day, of the pre-experiment period $t = 2, .., 14$, we calculate the key metric X over the time period that starts on this day t and finishes on the last day of the period. In this way, we obtain the cumulative time series of length 13. Actually, the cumulative time series is a set of features similar to **Total**, but calculated over the shorter pre-experiment periods of length from 1 to 13. The time series are known to be useful to improve the prediction quality of engagement metrics [10]. We refer to these 27 features as **TS**.

The cookie timestamps. Since a user's cookie may be created during an A/B experiment or some days before it, the information presented in the pre-experiment period (the above mentioned features) will not completely describe the actual behavior of the user. For instance, a user could be very active and could use the considered web service each day, but if she clears cookie files in her browser right before the experiment, a new cookie id will be assigned to her, and the number of sessions for this cookie id over 14 days before the experiment's period will be equal to zero, that will represent the user as an inactive one. Hence, in order to distinguish inactive users and users that cleared theirs cookies shortly before the experiment and, thus, assess the confidence in the information contained in the pre-experiment data, we consider, as features important for the prediction

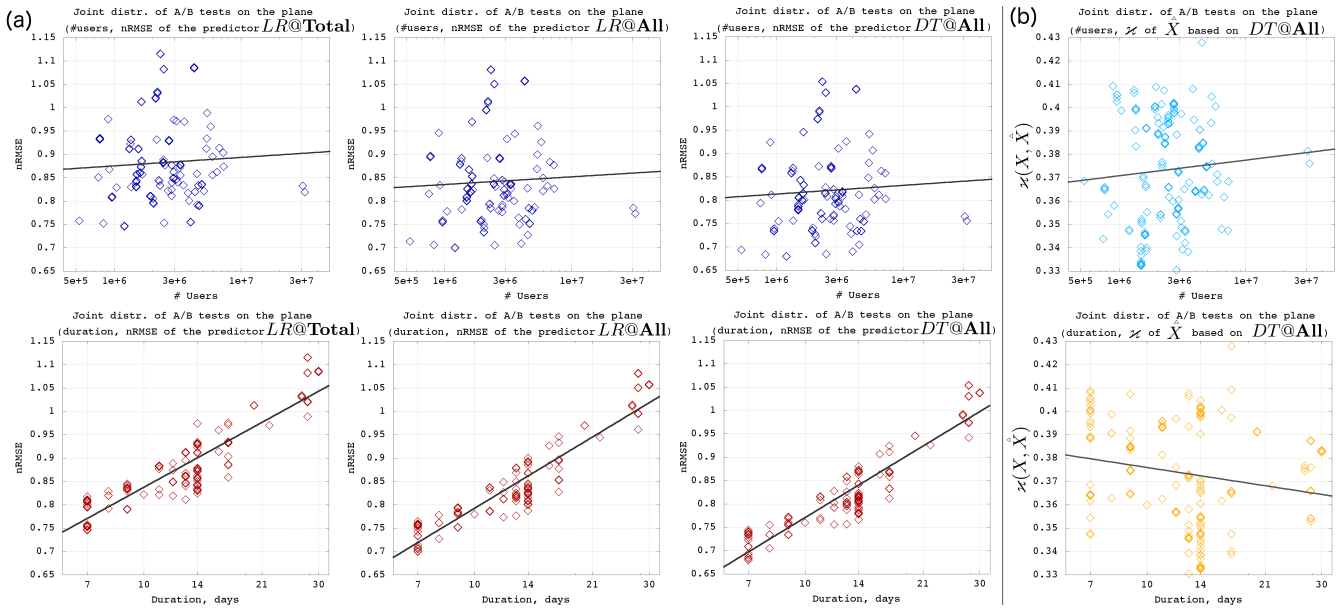

Figure 2: The joint distributions of 161 A/B experiments w.r.t.: (a) (their user traffic; predictor's nRMSE) and (their duration; predictor's nRMSE) for 3 prediction models (LR@Total**, LR@**All**, and DT@**All**); (b) (their user traffic; the VR rate $\varkappa$) and (their duration; the VR rate $\varkappa$) for the approach based on DT@**All**.**

task, the differences $t_e - t_c$, $t_f - t_e$, and $t_f - t_c$ (denoted by **CT**), where t_c is the creation time of the user cookie; t_f is the time of the first entrance[6] of the user in the experiment[7]; and t_e is the time of the experiment start, which is a constant for all users. Note that the time of the first entrance t_f does not depend on the treatment assignment (although, it is collected during the experiment's period), hence it does not violate the condition of independence of the predictor that is critical for Eq. (3).

The transformation features. The study [10] shows that the prediction quality could be noticeably improved by transformations of the daily time series from **TS**, namely, by the ones that reflect the periodicity (e.g., discrete Fourier amplitudes [9]) and the average amount of information (e.g., different variants of entropy [10]) of user engagement. We utilized 20 most profitable transformations according to the study [10]. We refer to these features as **TrTS**.

We define the set of all features described above by **All** = **Total** ∪ **TS** ∪ **CT** ∪ **TrTS**. Note that features **Total**, **TS**, and **TrTS** are different characteristics of the dynamics of the metric X. We could also use the same characteristics of some other metrics as features to better predict the target value of X. Investigation of this idea in [10] showed that, in the case of $X = S$, these characteristics of other user engagement metrics (considered in [10]) do not noticeably improve the prediction quality, while, in the case of $X = ATpS$, these characteristics of both $ATpS$ and S are useful. Hence, for the absence time per session as the target, we use both $ATpS$ and S to derive the features, while, for the number of sessions as the target, we use only S.

5.3 Models

We utilized two models to predict the values of the targets by minimizing the RMSE as the loss function.

[6]the first activity of the user since the start of the A/B test

[7]This time is more informative than the day of the first entrance (a categorical feature) used as a covariate in [7].

Linear model (LR). The first one is a classical linear regression model, which regards the prediction as an ordinary least square (OLS) problem. When the training set coincides with the data of a considered A/B test (i.e., we deal with a local predictor, see Sec. 3.3), our variance reduction approach in the case of this model becomes the classical linear regression adjustment on a set of covariates that is applied in randomized experiments in clinical studies, social sciences, etc. [14, 28]. Therefore, we consider the linear prediction model built on the feature **Total** (i.e., the total number of sessions during the 14-day pre-experiment period) as our first baseline model (since it coincides with the one applied earlier in online experimentation [7]), and we consider the linear prediction model built on all available features as our second baseline (since it was previously used in randomized experiments [13, 14, 28, 25, 3], but never applied to large-scale *online* A/B tests).

Decision trees' model (DT). The second model is the state-of-the-art Friedman's gradient boosted decision trees [15]. In our experimentation (Sec. 5.4 and 6), we use a proprietary implementation of the machine learning algorithm with 100 iterations and 100 trees, where features were processed by means of the equal frequency binning with 64 bins.

We use the short notation M@**FS** for a model M built on features **FS** (e.g., DT@**All** denotes decision trees built on the features **All** and LR@**Total** denotes the linear regression model based on the feature **Total**).

5.4 Prediction quality

In order to validate the quality of our predictors, we utilize all our 161 A/B experiments. For each experiment, we measure the prediction quality in terms of the normalized RMSE (nRMSE) defined by $\mathrm{nRMSE}(X, \widetilde{X}) := \mathrm{RMSE}(X, \widetilde{X})/\mathrm{E}(X)$ for the actual target X and the predictor $\widetilde{X}$. This allows us, first, to hide the information about the magnitude of the studied metric for confidentiality reasons, and, second, to make the quality measure independent from this magnitude

(e.g., when we report the average values), since it significantly depends on the durations of an experiment [9, 10].

Comparison of features and models. In Table 1, we report the average value of the nRMSE for each local prediction model based on different feature sets over all our A/B experiments. All differences between the presented nRMSE are statistically significant with p-value $\leq 10^{-3}$ (measured by paired t-test over the A/B experiments). First, we see that the best predictor is decision trees which is based on the set of all features. Second, our novel timestamps features **CT** noticeably improve the prediction quality w.r.t. **Total** $\cup$ **TS**: by 1.31% for the linear model and by 3.55% for decision trees. The features **CT** carry information about different types of users and, thus, resemble categorical features, so the decision trees may better utilize these features than the linear model. Third, the transformation features **TrTS** improve the prediction quality as well. Note that this improvement is around of 0.11% for decision trees, hence, for this model, one can use the set of features **Total** $\cup$ **TS** $\cup$ **CT** in order to reduce the computational complexity without a critical loss in prediction quality. Summarizing, we conclude that *decision trees built on all features (DT@**All**) significantly outperforms our baselines: LR@**Total** by* 7.2% *and LR@**All** by* 2.55%.

Duration and user traffic. We study the dependence of the prediction quality on the duration of an experiment and on the number of users participated in it. In Figure 2 (a), we report the joint distributions of our 161 A/B experiments w.r.t. the nRMSE and each of these two quantities in the logarithmic scale, due to the space constraints, only for the baseline predictors *LR*@**Total**, *LR*@**All** and for the best one *DT*@**All**. First, we see that the prediction quality only weakly depends on the size of user samples: the slope of the best fit line is very low (from 0.007 to 0.008) and its standard error (SE) is hight ($\approx$ 0.01). It is an expected result, since the prediction quality should not depend on the size of the training and test data, when their sizes (in our case, at least hundreds of thousands users) are large enough w.r.t. the number of features (in our case, no more than a hundred). Second, the prediction quality in terms of the nRMSE clearly linearly depends on the duration of the experiment: the slope of the best fit line is high (from 0.187 to 0.206) and its SE is very low (from 0.006 to 0.007). Thus, we conclude that *the longer an experiment, the worse the quality of prediction*, which is expected since the pre-experiment data of a user becomes more uncertain about her future behavior.

Local vs. global predictor. The results presented above are obtained for local predictors, i.e., the models are trained on all user traffic $\mathcal{U}$[8] for each experiment individually (see Sec. 3.3 for details). In order to understand the advantage of the utilization of a local training w.r.t. to a global one, first, we use the dataset collected from 2013 (described in Sec. 5.1) as the training set for a global predictor. Second, we truncate the duration of all experiments to one week and filter out A/B tests from 2013 year, obtaining a reduced set of 146 experiments of 2014 year. Thus, these experiments occurred later than the training data and have the same duration as the length of the target period of the training examples. On these A/B tests, we compare the nRMSE of the local predictor *DT*@**All** and the global one, which is based on the same learning model and the same feature set, but trained on the above described training set from 2013 year. The average nRMSE of the global predictor is 0.755, while the one of the local predictor is 0.726, which is lower by 3.96%. Thus, we conclude that *a predictor, trained on the data from the experiment's period, definitely outperforms the predictor with the same model and the same set of features, but trained on a dataset collected from a far earlier period.* This has been expected, since user behavior significantly depends on a time period, where this behavior is considered [9] (see Sec. 3.3).

6. A/B EXPERIMENTATION

In the context of variance reduction (VR) and sensitivity improvement, we consider 3 main baseline methods for our approach. The first one is the "zero" baseline, which is our source metric without any modification (e.g., the number of sessions in Sec. 6.1 and 6.2). The other two baseline methods are simplified versions of our approach: one technique coincides with the one applied earlier in online experimentation [7] (it is based on *LR*@**Total**), and the other is its extension based on the practice of randomized experiments in clinical and social studies [14, 28] (it is based on *LR*@**ALL**).

6.1 Variance reduction

We remind, see Sec. 3.2, that the performance of a variance reduction method is measured in terms of the reduction rate $\varkappa(X, \hat{X}) := \mathrm{Var}(\hat{X})/\mathrm{Var}(X)$, where X is the source key metric and $\hat{X}$ is the modified one by the method.

Comparison of features and models. In Table 2, we report the average value of the variance reduction rate $\varkappa$ for each local prediction model[9] based on different feature sets over all our A/B experiments. All differences between the presented VR rates are statistically significant with p-value $\leq 10^{-3}$ (measured as in Sec. 5.4). First, we see that the best VR method is the one which utilizes decision trees based on the set of all features. *It achieves* 62.66% *of saved user traffic on average* (see Eq. (2) and Sec. 3.2). Second, each set of features demonstrates a significant profit in terms of variance reduction: the time series **TS** and our novel timestamps features **CT** noticeably improve the variance reduction rate $\varkappa$ (e.g., for **CT**, by 2.76% and by 7% for *LR* and *DT* models respectively); the transformation features **TrTS** have a lower but positive improvement as well. Third, note that, in all cases except the one with one feature **Total**, decision trees has better performance than the one of the linear regression based on the same set of features. Overall, we conclude that *decision trees built on all features (DT@**All**) significantly outperforms our baselines: LR@**Total** by* 13.9% *and LR@**All** by* 5.1% *in terms of the variance reduction $\varkappa$ and, hence, in terms of saved user traffic.*

Duration and user traffic. We study the dependence of the variance reduction rate on the duration and the user traffic size of an experiment. In Fig. 2 (b), we plot the joint distributions of our 161 A/B experiments w.r.t. the rate $\varkappa$ and one of these quantities in the logarithmic scale. Due to space constraints, the results are presented only for the best method, which is based on *DT*@**All**, but they are similar for

[8]We considered different user sets as train data (including the control or the treatment user set solely) for local predictors, but the prediction quality was not noticeably different.

[9]Local predictors are better than global ones in terms of nRMSE (see Sec. 5.4) and in terms of $\varkappa$ (e.g., relative difference of $\varkappa$ for *DT*@**All** is 4.71%).

Table 2: Comparison of feature sets and models in terms of the average VR rate $\varkappa$ over 161 A/B test (relative improvement w.r.t. the previous row).

Feature set \ Model	Linear Regression		Decision Trees	
The source metric	1	(0%)	1	(0%)
Total	0.4337	(−56.63%)	0.4481	(−55.19%)
Total ∪ TS	0.4108	(−5.27%)	0.4046	(−9.7%)
Total ∪ TS ∪ CT	0.3995	(−2.76%)	0.3743	(−7.49%)
All	0.3935	(−1.5%)	**0.3734**	(−0.25%)

Table 3: Comparison of feature sets and models in terms of the success sensitivity rate (with $\alpha = 0.05$) over 161 A/B tests.

The source metric:	12 (7.45%)	
Feature set \ Model	Linear Regression	Decision Trees
Total	17 (10.55%)	18 (11.18%)
Total ∪ TS	22 (13.66%)	21 (13.04%)
Total ∪ TS ∪ CT	19 (11.80%)	**24** (**14.91**%)
All	20 (12.42%)	**24** (**14.91**%)

all other methods. First, we see that the variance reduction weakly depends on the size of user samples (the slope of the best fit line ≈ 0.003 with SE ≈ 0.0029). Second, the variance reduction is weakly depends on the duration of the experiment as well (the slope of the best fit line ≈ -0.01 with SE ≈ 0.0043). Thus, we conclude that *the variance reduction rate of our methods weakly depend on the user traffic size and the duration of an experiment.* The last result, together with the dependence of nRMSE on the duration (see Sec. 5.4), implies that this dependence of nRMSE is caused by the dependence of the z-score $z(X) := E(X)/\sqrt{\mathrm{Var}(X)}$ on the duration, since $\mathrm{nRMSE}^2(X, \widetilde{X}) = \varkappa(X, \hat{X})/z^2(X)$ for an unbiased predictor $\widetilde{X}$ (see Eq. (5)). The decrease of our z-score $z(X)$ with the growth of the experiment duration is a reproducing of the well known property of the number-of-sessions metric [19].

Matching. For the matching estimator $\Delta_{match}(X)$ defined by Eq. (7) and described in Sec. 3.3, we obtain the variance reduction rate equal to $\varkappa = 0.3749$. Hence, the relative improvement of the rate $\varkappa$ of the estimator $\Delta(\hat{X})$ based on the predictor DT@**All** w.r.t. the one of the matching estimator $\Delta_{match}(X)$ is equal to 0.39%. Thus, we conclude that *our approach based on the classical form of the estimator $\Delta(\hat{X})$ outperforms the matching estimator $\Delta_{match}(X)$ with matching based on the same model (i.e., decision trees) and the same set of features (i.e.,* **All***).*

6.2 Sensitivity improvement

Control of false-positive rates. In A/B testing, correctness of an experimentation is verified by A/A tests (i.e., control experiments) [23, 4]. Each of them compares two identical variants of the service. If a considered OAC (i.e., an OEC with a statistical test, see Sec. 3.1) is valid, then the p-value of this OAC should be uniformly distributed over $[0, 1]$ on A/A tests and the A/A tests should fail in not more than 5% of cases for the p-value threshold $\alpha = 0.05$ [23, 4, 11]. The number of failed A/A tests is referred to as the *false-positive rate* (also known as the type I error). We obtain a thousand of A/A experiments (like in [2, 7, 11]) by randomly splitting users from the control group of one of our A/B experiments. All our source and modified metrics do not fail the predefined false-positive rate threshold.

Success sensitivity rates. Following [9, 10, 8, 29, 11], we compare the sensitivity of our OACs in terms of the *success sensitivity rate*, i.e., the number of A/B tests whose treatment effect is detected by an OAC. In Table 3, we present these rates for our source metric X (i.e., the number of sessions) and for the metrics $\hat{X}$, modified by our approach based on different prediction models and sets of features over all our A/B experiments. We see that *the best variance reduction method, i.e., the one based on DT@**All**, has the best sensitivity improvement: the corresponding OAC outperforms the OAC of the source metric by increasing the success sensitivity rate twice and the one of the OAC with LR@**All** (baseline VR method) by* 20% (the same improvement is achieved by DT@**All** $\setminus$ **TrTS**).

6.3 Other metrics

In this subsection, we report the results of applying our best variance reduction method (i.e., the one based on DT@**All**) to the absence time metric $ATpS$, the other popular engagement metric of user loyalty [12] (see its definition in Sec. 4). We also apply two ways of filtering out users. In the first filter (s-filter), a user with only one session during the experiment period is removed from the user sample $\mathcal{U}$ (as in [11]). The second filter (t-filter) removes any user, who has a browser cookie created later than 24 hours before the experiment start (i.e., the cookie is very "young"). These filters are expected to improve the sensitivity of the source metric, since the removed users are believed to be less affected by the treatment effect.

We report the variance reduction rates $\varkappa$ and the success sensitivity rates in Table 4. The percent of variance reduction is reported relative to the source metric without any user filter. Thus, we are able to understand *the cumulative variance reduction rate* $\varkappa_c$ resulted from applying a user filter and the variance reduction technique together: $\varkappa_c = \varkappa(X, \hat{X}_f) = \varkappa(X, X_f) \cdot \varkappa(X_f, \hat{X}_f)$, where X_f is the source metric X over filtered users. First, the results for the number of sessions S and for the absence time $ATpS$ are similar. Second, the best cumulative variance reduction rate is demonstrated by our technique without any filter, while the best $\varkappa$ is demonstrated for the metrics with the t-filter. Note that the increase in variance is expected when we apply the filters, since the removed users have similar or even identical behavior. The user filters really improve the sensitivity of the source metric X, but their combination with the variance reduction technique is not better than the variance reduction technique applied solely. Thus, we conclude that *our technique based on decision trees and all available features noticeably reduces the variance of all engagement metrics of user loyalty (from* 62.66% *to* 58.48% *depending on applied filters) and improves their sensitivity up to twice.*

7. CONCLUSIONS AND FUTURE WORK

In our work, we focused on the problem of variance reduction for engagement metrics of user loyalty that are widely used in A/B testing of web services. We developed a general framework that is based on machine learning techniques, that allowed us, on the one hand, to perform a deep study of existing approaches used in randomized experiments in online and offline studies (like clinical trials), and, on the other hand, to propose a new class of methods based on ensembles of decision trees, that, to the best of our knowledge, have

Table 4: Comparison of source metrics X and their modifications $\hat{X}$ (based on DT@All) in terms of the average VR rate $\varkappa$ (relative improvement w.r.t. non-filtered X) and the success sensitivity rate (with $\alpha = 0.05$) over 161 A/B tests.

Metric $X =$	Variance reduction rate $\varkappa(X, \hat{X})$	Success sensitivity rate for X	Success sensitivity rate for $\hat{X}$
S	0.3734 (**−62.66**%)	12 (7.45%)	**24** (**14.91**%)
S (s-filter)	0.4152 (−32.07%)	16 (9.94%)	21 (13.04%)
S (t-filter)	**0.3668** (−33.42%)	15 (9.32%)	21 (13.04%)
$ATpS$	0.3735 (**−62.65**%)	12 (7.45%)	**23** (**14.29**%)
$ATpS$ (s-filter)	0.4152 (−32.07%)	16 (9.94%)	22 (13.66%)
$ATpS$ (t-filter)	**0.3668** (−33.42%)	15 (9.32%)	20 (12.42%)

not been applied earlier to problems of variance reduction. We experimented with a very large and diverse set of 161 real large-scale A/B experiments. First, we have shown that our novel variance reduction technique (which is based on decision trees) outperformed state-of-the-art ones. Second, this technique demonstrated 63% average variance reduction (5.1% improvement over the best state-of-the-art technique), which is equivalent to 63% overall (5.1% relative) saved user traffic utilized in online evaluation. Finally, we also applied the method to sensitivity improvement that resulted in the detection of the treatment effect in 2 times more A/B tests than with non-modified user engagement metrics and +20% more A/B tests than the one modified by the state-of-the-art VR technique. Hence, our study produces essential results on effectiveness of different variance reduction techniques applied to user engagement metrics that coincide with the emerging needs of modern Internet companies to run more controlled experiments on a limited number of their users.

Future work. First, we can improve the prediction quality by more complicated models and richer feature sets for further variance reduction and sensitivity improvement. Second, one can further study matching estimators to achieve better sensitivity.

8. REFERENCES

[1] S. Addelman. The generalized randomized block design. *The American Statistician*, 23(4):35–36, 1969.

[2] E. Bakshy and D. Eckles. Uncertainty in online experiments with dependent data: An evaluation of bootstrap methods. In *KDD'2013*, pages 1303–1311, 2013.

[3] A. Bloniarz, H. Liu, C.-H. Zhang, J. Sekhon, and B. Yu. Lasso adjustments of treatment effect estimates in randomized experiments. *arXiv:1507.03652*, 2015.

[4] T. Crook, B. Frasca, R. Kohavi, and R. Longbotham. Seven pitfalls to avoid when running controlled experiments on the web. In *KDD'2009*, pages 1105–1114, 2009.

[5] A. Deng and V. Hu. Diluted treatment effect estimation for trigger analysis in online controlled experiments. In *WSDM'2015*, pages 349–358, 2015.

[6] A. Deng, T. Li, and Y. Guo. Statistical inference in two-stage online controlled experiments with treatment selection and validation. In *WWW'2014*, 2014.

[7] A. Deng, Y. Xu, R. Kohavi, and T. Walker. Improving the sensitivity of online controlled experiments by utilizing pre-experiment data. In *WSDM'2013*, 2013.

[8] A. Drutsa. Sign-aware periodicity metrics of user engagement for online search quality evaluation. In *SIGIR'2015*, pages 779–782, 2015.

[9] A. Drutsa, G. Gusev, and P. Serdyukov. Engagement periodicity in search engine usage: Analysis and its application to search quality evaluation. In *WSDM'2015*, pages 27–36, 2015.

[10] A. Drutsa, G. Gusev, and P. Serdyukov. Future user engagement prediction and its application to improve the sensitivity of online experiments. In *WWW'2015*, pages 256–266, 2015.

[11] A. Drutsa, A. Ufland, and G. Gusev. Practical aspects of sensitivity in online experimentation with user engagement metrics. In *CIKM'2015*, 2015.

[12] G. Dupret and M. Lalmas. Absence time and user engagement: evaluating ranking functions. In *WSDM'2013*, pages 173–182, 2013.

[13] D. A. Freedman. On regression adjustments to experimental data. *Advances in Applied Mathematics*, 40(2):180–193, 2008.

[14] D. A. Freedman, D. Collier, J. S. Sekhon, and P. B. Stark. *Statistical models and causal inference: a dialogue with the social sciences*. Cambridge University Press, 2010.

[15] J. H. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, 2001.

[16] H. Hohnhold, D. O'Brien, and D. Tang. Focusing on the long-term: It's good for users and business. In *KDD'2015*, pages 1849–1858, 2015.

[17] E. Kharitonov, C. Macdonald, P. Serdyukov, and I. Ounis. Optimised scheduling of online experiments. In *SIGIR'2015*, pages 453–462, 2015.

[18] E. Kharitonov, A. Vorobev, C. Macdonald, P. Serdyukov, and I. Ounis. Sequential testing for early stopping of online experiments. In *SIGIR'2015*, pages 473–482, 2015.

[19] R. Kohavi, A. Deng, B. Frasca, R. Longbotham, T. Walker, and Y. Xu. Trustworthy online controlled experiments: Five puzzling outcomes explained. In *KDD'2012*, 2012.

[20] R. Kohavi, A. Deng, B. Frasca, T. Walker, Y. Xu, and N. Pohlmann. Online controlled experiments at large scale. In *KDD'2013*, pages 1168–1176, 2013.

[21] R. Kohavi, A. Deng, R. Longbotham, and Y. Xu. Seven rules of thumb for web site experimenters. In *KDD'2014*, 2014.

[22] R. Kohavi, R. M. Henne, and D. Sommerfield. Practical guide to controlled experiments on the web: listen to your customers not to the hippo. In *KDD'2007*, pages 959–967, 2007.

[23] R. Kohavi, R. Longbotham, D. Sommerfield, and R. M. Henne. Controlled experiments on the web: survey and practical guide. *DMKD*, 18(1):140–181, 2009.

[24] P. L'Ecuyer. Efficiency improvement and variance reduction. In *Proceedings of the 26th conference on Winter simulation*, pages 122–132. SCSI, 1994.

[25] W. Lin et al. Agnostic notes on regression adjustments to experimental data: Reexamining freedman's critique. *The Annals of Applied Statistics*, 7(1):295–318, 2013.

[26] D. F. McCaffrey, G. Ridgeway, and A. R. Morral. Propensity score estimation with boosted regression for evaluating causal effects in observational studies. *Psychological methods*, 9(4):403, 2004.

[27] K. L. Morgan, D. B. Rubin, et al. Rerandomization to improve covariate balance in experiments. *The Annals of Statistics*, 40(2):1263–1282, 2012.

[28] S. L. Morgan and C. Winship. *Counterfactuals and causal inference*. Cambridge University Press, 2014.

[29] K. Nikolaev, A. Drutsa, E. Gladkikh, A. Ulianov, G. Gusev, and P. Serdyukov. Extreme states distribution decomposition method for search engine online evaluation. In *KDD'2015*, pages 845–854, 2015.

[30] Y. Song, X. Shi, and X. Fu. Evaluating and predicting user engagement change with degraded search relevance. In *WWW'2013*, pages 1213–1224, 2013.

[31] D. Tang, A. Agarwal, D. O'Brien, and M. Meyer. Overlapping experiment infrastructure: More, better, faster experimentation. In *KDD'2010*, pages 17–26, 2010.

[32] J. R. Wilson. Variance reduction techniques for digital simulation. *American Journal of Mathematical and Management Sciences*, 4(3-4):277–312, 1984.