Online Assortment Optimization with High-Dimensional Data

Xue Wang* Mike Mingcheng Wei* Tao Yao†

*Penn State University, Industrial and Manufacturing Engineering, xzw118@psu.edu
*University at Buffalo, School of Management, mcwei@buffalo.edu
†Penn State University, Industrial and Manufacturing Engineering, tyy1@engr.psu.edu

In this research, we consider an online assortment optimization problem, where a decision-maker needs to sequentially offer assortments to users instantaneously upon their arrivals and users select products from offered assortments according to the contextual multinomial logit choice model. We propose a computationally efficient Lasso-RP-MNL algorithm for the online assortment optimization problem under the cardinality constraint in high-dimensional settings. The Lasso-RP-MNL algorithm combines the Lasso and random projection as dimension reduction techniques to alleviate the computational complexity and improve the learning and estimation accuracy under high-dimensional data with limited samples. For each arriving user, the Lasso-RP-MNL algorithm constructs an upper-confidence bound for each individual product's attraction parameter, based on which the optimistic assortment can be identified by solving a reformulated linear programming problem. We demonstrate that for the feature dimension d and the sample size dimension T, the expected cumulative regret under the Lasso-RP-MNL algorithm is upper bounded by $\tilde{\mathcal{O}}(\sqrt{T}\log d)$ asymptotically, where $\tilde{\mathcal{O}}$ suppresses the logarithmic dependence on T. Furthermore, we show that even when available samples are extremely limited, the Lasso-RP-MNL algorithm continues to perform well with a regret upper bound of $\tilde{\mathcal{O}}(T^{\frac{2}{3}}\log d)$. Finally, through synthetic-data-based experiments and a high-dimensional XianYu assortment recommendation experiment, we show that the Lasso-RP-MNL algorithm is computationally efficient and outperforms other benchmarks in terms of the expected cumulative regret.

Key words: Online assortment optimization, contextual information, high-dimensional data, Lasso, random projection, multinomial logit model, upper-confidence bound.

1. Introduction

Online assortment optimization problems have recently emerged in many internet applications, such as e-tailing, digital advertising, recommendation, etc., where a decision-maker sequentially offers assortments of substitutable products to users instantaneously upon their arrivals. For example, an internet retailer offers an assortment of customized items to an arriving consumer; a search engine displays an assortment of profitable advertisements in response to a user's search query; an online marketplace recommends an assortment of personalized products based on a user's browsing and purchasing history. To select reward-maximizing assortments, the decision-maker first needs

to accurately assess users' utilities and choices, which are typically unknown a priori but can be gradually learned through observing users' responses to various assortments (Mahajan and van Ryzin 1999). Hence, online assortment optimization problems require a judicious balance between exploring different assortments to learn users' choices and simultaneously exploiting assortments that maximize immediate rewards. In the big data era, the growing availability of granular data and high-dimensional contextual information for users and products has presented both promising opportunities and vexing challenges for these online assortment optimization problems.

Rich contextual information provides the decision-maker with unprecedented opportunities to improve his learning capability and prediction accuracy regarding users' utilities and choices. Consider the assortment recommendation practice at XianYu, a leading consumer-to-consumer online marketplace for new and preowned products in China. Upon clicking a product link, the user is redirected to the product information page, on which, along with typical product specifics and transaction details, a "Guess What You Like" section displays a personalized assortment consisting up to 20 suggested products. To optimally recommend 20 products, XianYu relies on contextual information about the user and products to learn and predict the user's utility and choice concerning any given assortment. Realizing that more information leads to better learning and prediction, XianYu has dramatically expanded the extent of contextual information and accelerated its data collection efforts. Currently, the available contextual information at XianYu is extremely high-dimensional: It contains more than 2 billion features, including user information (e.g., demographics, geographics, browsing/clicking history, etc.), product information (e.g., brand, color, size, condition, etc.), and information about possible interactions between these two (e.g., the physical distance between the user and the product, whether the user's searching history matches the product, etc.). In practice, using this high-dimensional contextual information has significantly improved XianYu's learning and prediction accuracy regarding users' choices, which in turn enables better assortment decisions.

Yet, the decision-maker's ability to use all available contextual information and effectively learn the influences of all features on users' utilities and choices is often impaired by the fact that there are limited samples in practice. Specifically, to accurately estimate the influences of more than 2 billion features via traditional statistical methods (e.g., maximum likelihood estimation), XianYu will need billions or even trillions random samples. However, with approximately 4 million daily "Guess What You Like" exposures, among which a very small percentage can be chosen to perform costly learning experiments (Bastani and Bayati 2015), it may take decades before XianYu can identify the influences of all features with reasonable accuracy. Moreover, due to the intrinsically ever-changing nature of users' tastes, these estimations are typically time sensitive: Estimations based on historical data stretching more than a year, or a few months for fashionable

products, will be less relevant for predicting users' choices tomorrow. Hence, compared to the scale of high-dimensional contextual information, available samples are extremely limited and therefore constrain the decision-maker's ability to fully utilize all available features to learn and update his estimations.

Furthermore, even with sufficient samples to support effective learning, the decision-maker still has to ensure that the online assortment optimization algorithm is computationally efficient. In XianYu's example, the average time that elapses between a user clicking a product link and the web page displaying the recommended assortment is expected to be less than a half-second, which includes time needed for learning/updating the estimations and optimizing recommended assortments. However, a single estimation update for XianYu's high-dimensional features can easily take hours with high-performance computing technologies and state-of-the-art techniques, especially when the sample size is not too small; combined with the time needed for optimizing assortments, which is a nonlinear combinatorial optimization problem, the total computational time may far exceed the half-second target mark.

To address these challenges, we propose a computationally efficient Lasso-RP-MNL algorithm for online assortment optimization problems under the cardinality constraint in high-dimensional settings. This algorithm combines both the Lasso (Tibshirani 1996) and random projection (Johnson and Lindenstrauss 1984) to improve the learning and estimation accuracy for high-dimensional features with limited samples and follows the idea of upper-confidence bound (UCB) approach (Auer 2002) to identify the optimistic assortment under the multinomial logit (MNL) choice model. In particular, with a period length that exponentially increases in time, we periodically uses the Lasso to identify and update significant features that have strong influences on users' utilities and choices; then, for each arriving user, we adopt random projection to reduce the high-dimensional contextual information, excluding significant features already identified by the Lasso, to a low-dimensional space and then estimate coefficients for both original features identified by Lasso and projected features by random projection. Through this process, the learning and parameter estimation can be performed in a low-dimensional fashion to significantly trim down the computational time, while maintaining high accuracy in predicting users' choices. Furthermore, we show that using the Lasso for feature selection will limit the long-term negative influence of the information loss that is intrinsic to random projection and that random projection can in turn alleviate the negative influence of possible model misspecification in the Lasso due to limited samples. Next, we construct the upper-confidence bound for each individual product's utility and then identify the optimistic assortment by solving a reformulated linear programming problem. Finally, we develop a random decay sampling schedule, which generates sufficient random samples to accelerate the learning process in the data-poor regime and then reduces to a lower level to improve the regret

performance in the data-rich regime. To the best of our knowledge, our study is the first to propose a computationally efficient online assortment optimization algorithm for high-dimensional data under limited samples.

We theoretically demonstrate that the Lasso-RP-MNL algorithm can significantly improve the cumulative regret upper bound in high-dimensional settings and achieve a logarithmic dependence on the feature dimension. Specifically, the expected cumulative regret of the Lasso-RP-MNL algorithm is asymptotically upper-bounded by $\tilde{\mathcal{O}}(\sqrt{T}\log d)$, where T and d are the time/sample dimension and the feature dimension respectively and we use $\tilde{\mathcal{O}}$ to suppress the logarithmic dependence on T. Compared to UCB-type bounds in Chen et al. (2018) and Oh and Iyengar (2019a) or Thompson-Sampling bounds in Oh and Iyengar (2019b), the Lasso-RP-MNL algorithm improves the linear or sublinear dependence on the feature dimension d to a logarithmic dependence. Such an improvement is particularly significant for the regret performance under high-dimensional settings, where the feature dimension is much larger than the sample size dimension. In addition, we show that the Lasso-RP-MNL algorithm continues to perform well even under the data-poor regime, where the available samples are extremely limited, with the cumulative regret upper bound $\tilde{\mathcal{O}}(T^{\frac{2}{3}}\log d)$. We believe that the Lasso-RP-MNL algorithm is the first algorithm in high-dimensional settings to attain logarithmic dependence on the feature dimension for online assortment optimization problems.

Finally, we benchmark the Lasso-RP-MNL algorithm to existing state-of-the-art algorithms in the literature and industrial practice through both synthetic experiments and a experiment based on XianYu's high-dimensional assortment recommendation dataset. In the first synthetic experiment, we explore the benefits of the Lasso-RP-MNL algorithm by comparing it to four benchmarks (i.e., MNL-Bandit by Agrawal et al. 2019 and three other ancillary algorithms) to illustrate the value of incorporating contextual information and the value of adopting dimension reduction techniques (i.e., the Lasso and random projection). Then, in the second synthetic experiment, we simulate the real practice environment, where the decision-maker faces high-dimensional contextual information and a large product candidate set that varies from user to user, to examine the impacts of system parameters on the Lasso-RP-MNL algorithm's cumulative regret performance and computational time. We observe that the Lasso-RP-MNL algorithm is computationally efficient and generates the lowest regret in all synthetic experiments. Finally, we adopt XianYu's high-dimensional assortment recommendation dataset and benchmark the Lasso-RP-MNL algorithm to XianYu's Top-K algorithm and two other feasible benchmarks. In this last experiment, we observe that the Lasso-RP-MNL algorithm continues to be computationally efficient for real-life large-scale problems and can significantly improve the decision-maker's revenue performance.

2. Related Literature

Our work is related to the dynamic assortment optimization literature, where users' utilities (i.e., model parameters) are unknown to the decision-maker at the beginning but can be gradually learned over multiple periods. Various models have been used in the assortment optimization literature, such as the MNL model (e.g., Ryzin and Mahajan 1999, Mahajan and Van Ryzin 2001), nested logit (e.g., McFadden 1980, Gallego and Wang 2014, Li et al. 2015), exogenous demand model (e.g., Smith and Agrawal 2000, Netessine and Rudi 2003), Markov chain model (e.g., Blanchet et al. 2016, Feldman and Topaloglu 2017), and non-parametric models (e.g., Rusmevichientong et al. 2006, Farias et al. 2013). Among these models, the MNL model, which is adopted in this work, is the most commonly used choice model in Economics, Marketing, and Operations Management literature (Kök and Fisher 2007), mainly by virtue of its tractability in estimating unknown parameters and identifying optimal assortments. For extensive literature review on the MNL model and other assortment optimization models, we refer to Mahajan and van Ryzin (1999) and Kök et al. (2015).

When there are limited number of products repetitively offered to incoming users, it is natural to consider the setting where the utility of each product is represented by an unknown attraction parameter in the MNL model. Under this setting, Rusmevichientong et al. (2010) and Sauré and Zeevi (2013) propose two explore-then-exploit algorithms, where the decision-maker first offers pre-selected assortments in the exploration phase to attend desired estimation accuracy for these unknown parameters, and then goes to the exploitation phase to maximize his expected reward. Rusmevichientong et al. (2010) show that their Adaptive Assortment algorithm can attain $\mathcal{O}(N^2 \log^2 T)$ cumulative regret bound, and Sauré and Zeevi (2013) demonstrate that their separation-based policy can achieve $\mathcal{O}(N \log T)$ regret ratio bound, where N is the number of candidate products. Kallus and Udell (2016) consider a personalized assortment model to extend the homogeneous users case to the heterogeneous case, and Bernstein et al. (2018) adopt a Bayesian semi-parametric framework to propose a dynamic clustering policy to map users' profiles to groups/clusters. It is worth noting that these works require certain a prior knowledge of the separation gap parameter, which gauges the reward difference between the optimal and the second-best assortment to regulate the exploration phase, without which these algorithms can perform quite poorly (Agrawal et al. 2019).

Without assuming any prior knowledge of the separation gap parameter, Agrawal et al. (2019) propose MNL-Bandit. In their work, to obtain unbiased estimate for each product's utility, the authors offer an assortment repeatedly until a no-purchase occurs and use the sample mean of times a product is purchased as the estimate. Then, the authors develop a UCB-type policy and

identify the optimistic assortment by solving a nonlinear optimization problem. Under the assumption that the no-purchase is the most "frequent" outcome, the authors establish the regret upper bound $\tilde{\mathcal{O}}(\sqrt{NT})$. Agrawal et al. (2017) further propose another algorithm based on the Thompson-Sampling approach that can also achieves the same regret bound with improved empirical performance. Different from these two papers that study homogeneous users under the stochastic arrival setting, Cheung and Simchi-Levi (2017) study heterogeneous users under the adversarial user arrival. The authors propose a Thompson Sampling based Pao-Ts policy whose Bayesian regret upper bound satisfies $\tilde{\mathcal{O}}(N\sqrt{T})$.

All of previous works assume that unknown parameters are associated with products themselves (i.e., each product has an unique unknown attraction parameter). In practice, however, the number of products can be enormous, and available products change from user to user, both of which lead to an unnecessarily large number of unknown parameters needed to be learned. Therefore, recognizing the facts that the difference among products can be represented by their intrinsic features and that a smaller number of features are sufficient to identify a large number of products in practice (Agrawal et al. 2019), the contextual MNL model assigns unknown parameters to every unique feature and estimates these feature parameters separately. As features can be shared among multiple products, the learning can now cross products (Oh and Iyengar 2019a), which suggests that the regret bound for the contextual MNL model can be independent of the number of candidate products N.

Chen et al. (2018) consider the contextual MNL model in which the feature information of products can change over time (i.e., the underlying choice model is non-stationary) and develop an explore-then-exploit UCB-based policy with $\tilde{\mathcal{O}}(d\sqrt{T})$ regret bound. The authors also establish a lower bound $\Omega(d\sqrt{T}/K)$ and argue that their policy is optimal up to logarithmic factors. Following a similar setting, Oh and Iyengar (2019a) propose another two explore-then-exploit UCB-based algorithms: The first computationally efficient algorithm attain the regret bound of $\tilde{\mathcal{O}}(d\sqrt{T})$, and the second algorithm reduces the regret bound to $\tilde{\mathcal{O}}(\sqrt{dT})$. Oh and Iyengar (2019b) develop two Thompson sampling algorithms and achieve $\tilde{\mathcal{O}}(d^{3/2}\sqrt{T})$ and $\tilde{\mathcal{O}}(d\sqrt{T})$ Bayesian regret, respectively. Yet, a polynomial, linear, or sublinear dependence on the feature dimension d hinders these algorithms from practically implementing for online assortment optimization problems under high-dimensional data settings, mainly due to dissatisfied regret performance and the excessive computational burden. In this work, we combine both the Lasso and random projection to develop a simultaneously-explore-and-exploit Lasso-RP-MNL algorithm that is computationally efficient and improve the regret bound to $\tilde{\mathcal{O}}(\sqrt{T}\log d)$ asymptotically or to $\tilde{\mathcal{O}}(T^{\frac{2}{3}}\log d)$ with very limited samples. We believe that the Lasso-RP-MNL algorithm is the first online assortment optimization algorithm in high-dimensional settings to attain logarithmic dependence on the feature dimension.

At last, as our algorithm combines the Lasso and random projection to handle the high-dimensional data challenges, this paper is also related to these two streams of literature. In high-dimensional statistics, Lasso-type methods (Tibshirani 1996) have been proposed to explore the high-dimensional data's underlying latent sparse structure and become a standard approach for high-dimensional feature selection and learning (Fan and Li 2001, Meinshausen et al. 2006, 2009, Zhang et al. 2010, Loh and Wainwright 2013). For example, Belloni et al. (2013) study the OLS post-Lasso estimator that first uses the Lasso for feature selection and then applies OLS for parameter estimation. The authors show that it performs strictly better than the Lasso and has the advantage of a smaller bias, even when the feature selection misses some parameters of the true model (i.e., model misspecification). Lee et al. (2016) propose a general approach to valid confidence intervals after model selection via the Lasso. Our algorithm also periodically adopts the Lasso for feature selection. Yet, the Lasso may suffer from model misspecification, especially under limited samples, and can be computationally challenging, therefore restraining these algorithms from being implemented directly in online settings. Random projection (Johnson and Lindenstrauss 1984) has been proposed as a computationally efficient method to deal with high-dimensional data (Fern and Brodley 2003, Pilanci and Wainwright 2015). Specifically, random projection is one of matrix sketching methods (Matoušek 2008, Luo et al. 2016, Ghashami et al. 2016, Clarkson and Woodruff 2017) that approximate a high-dimensional matrix by a more compact low-dimension one with certain approximation guarantee. Therefore, the estimation for unknown parameters and assessment of utilities can be completed in a low-dimensional fashion to significantly reduce the computational complexity (Vershynin 2010) with acceptable accuracy loss. Yet, the distortion and information loss intrinsic to random projection may lead to significant regret loss. In this work, we combine random projection and the Lasso to limit the information loss and to curb model misspecification, while maintaining the computational efficiency.

3. Problem Statement and Preliminaries

Consider a sequential decision-making process: At each time $t \in \{1, 2, ..., T\}$, a single user/consumer arrives, and the decision-maker then offers this user an assortment A_t from a candidate set containing N_t products indexed by $1, 2, ..., N_t$ (i.e., $A_t \subseteq \{1, 2, ..., N_t\}$). The number of available products and the product candidate set change frequently over time, because some products may be sold out and unavailable in the future, some new products can be added to the candidate set, or some products should be excluded from certain user groups according to legal/managerial policies. Therefore, at different times, the same product index may refer to different products. At each time, due to the cardinality constrain (e.g., limited display capacity), the decision-maker can offer at most K products to the user, $|A_t| \leq K$.

Users are heterogeneous, and the contextual information that characterizes both an arbitrary product i and the user arriving at time t – the user-product pair – is prescribed by a feature vector $x_{t,i} \in \mathbb{R}^d$. This feature vector is high-dimensional and includes rich contextual information about the user, the product, and possible interactions between these two. If the user chooses a product j from the offered assortment A_t, then the decision-maker will collect a reward $r_{t,j}$, which can take the form of click-through, gross merchandise volume, commission revenue, etc. Note that the user can also choose nothing from the assortment A_t; that is the no-purchase option, in which case the decision-maker receives zero reward.

We model the user's choice for a given assortment using the MNL model, a utility-based model first introduced by McFadden et al. (1973). For simplicity, we suppress the time index t, as long as doing so does not cause any misinterpretation. Under the MNL model, the user's utility associated with a product i, where product 0 represents the no-purchase option with a zero feature vector (i.e., $x_0 = \mathbf{0}$), is given by

$$U_i = x_i^T \beta^* + \zeta_i, \ i = 0, 1, 2, ..., N$$

where $x_i^T \beta^* \in \mathbb{R}$ denotes product i's mean utility and ζ_0, ζ_1, ..., ζ_N are i.i.d. random variables following a Gumbel distribution with location parameter 0 and scale parameter 1. Note that $\beta^* \in \mathbb{R}^d$ is the unknown true parameter/coefficient vector for contextual information but can be learned by observing users' choices as time progresses. The probability that the user will choose product j for a given assortment A can be directly derived (see Anderson et al. 1992) as follows:

$$p_{\beta^*, A}(j) = \begin{cases} e^{x_j^T \beta^*} / \left(1 + \sum_{i \in A} e^{x_i^T \beta^*}\right) & , \ j \in A \cup \{0\} \\ 0 & , \text{ otherwise} \end{cases}$$

Following the literature, we will refer $e^{x_i^T \beta^*}$ to as the attraction parameter for product i. To avoid trivial decisions, we assume that the feature vector, the coefficient vector, the choice probability, and the reward are bounded so that the maximum reward is also upper-bounded. Formally, we make the following technical assumption:

Assumption A.1 (MNL model): Given an assortment A, an coefficient vector β, and feature vectors x_i for $i \in A$, the probability that the user chooses product $j \in A$ follows the MNL model and is given by $p_{\beta, A}(j)$. Moreover, there exist positive constants b, $x_{\max}$, $R_{\max} \geq 1$, and ρ such that $\|\beta\| \leq b$, $\|x_i\|_1 \leq x_{\max}$, $p_{\beta, A}(j) \geq \rho$, and $r_i \in (0, R_{\max}]$ for any product i.

The high-dimensional feature vector can include all the information available to the decision-maker, but not all available features are equally valuable for predicting the user's utility and choice. For example, the user's age, the product's brand name, and the name of the product's distributor may all be available to the decision-maker and are included in the feature vector; among these three

features, the first two are typically more informative for assessing this user's utility and choice than the last one. Hence, in practice, the coefficient vector β^* naturally exhibits a latent sparse structure. Let $\mathcal{S}^* = \{j : \beta_j^* \neq 0\} \in \mathbb{R}^s$ denote the true index set for significant features (e.g., the user's age and the product's brand name), which have nonzero coefficient values and are therefore important bases for the decision-maker's predictions. The size of the index set is much smaller than the dimension of the feature vector (i.e., $s \ll d$), but the index set $\mathcal{S}^*$ itself is unknown to the decision-maker at the beginning.

The decision-maker's objective is to design a policy $\pi = \{A_t\}_{t\geq 1}$, where A_t is the assortment prescribed by policy π at time t, that maximizes the expected cumulative reward over T periods:

$$\sum_{t=1}^{T} \sum_{i \in A_t} r_{t,i} p_{\beta^*, A_t}(i).$$

Note that the true coefficient vector β^* is unknown to the decision-maker at the beginning, so it is generally intractable to directly analyze this equation. Instead, we benchmark the policy π to an oracle policy, where the decision-maker knows the true coefficient vector β^* and always picks the assortment that generates the highest expected reward. Specifically, we define the decision-maker's expected cumulative regret up to time T under the policy π as

$$\text{Regret}(T) = \sum_{t=1}^{T} \left\{ \max_{\substack{\tilde{A}_t \subseteq \{1,2,\ldots,N_t\} \\ |\tilde{A}_t| \leq K}} \left[\sum_{j \in \tilde{A}_t} r_{t,j} p_{\beta^*, \tilde{A}_t}(j) \right] - \sum_{i \in A_t} r_{t,i} p_{\beta^*, A_t}(i) \right\}, \tag{1}$$

which is the difference between the expected reward under the oracle policy and that under the current policy π. The decision-maker will explore to select the optimal policy π to minimize the expected cumulative regret.

4. The Lasso-RP-MNL Algorithm

In the section, we describe the Lasso-RP-MNL algorithm and establish its regret performance. We start with the learning and estimation of the unknown coefficient β^* using the Lasso and random projection. Specifically, §4.1 discusses the process of using the Lasso to learn the significant feature set $\mathcal{S}^*$ and demonstrates that this method can asymptotically recover significant features with high probability. §4.2 constructs the permutation matrix and the projection matrix to reduce the high-dimensional estimation problem into a low-dimensional space and shows that the coefficient vector under the proposed permutation and projection is nearly invariant. Moreover, we demonstrate that using the Lasso for feature selection will limit the long-term negative influence of the information loss that is intrinsic to random projection and that random projection can in turn alleviate the negative influence of possible model misspecification in the Lasso due to limited samples. Next, in §4.3, we construct the upper-confidence bound for each individual product's utility, identify

the optimistic assortment by solving a reformulated linear programming problem, and establish the single-period regret upper bound for this optimistic assortment. Finally, in §4.4, we formally present the Lasso-RP-MNL algorithm and derive its expected cumulative regret upper bound.

4.1. The Lasso and Feature Selection

Denote the observed users' choices, in response to assortments $\{A_1, A_2, ..., A_T\}$ up to time T, as $\{c_1, c_2, ..., c_T\}$. Further, among all previously offered assortments, we use $\mathcal{W}_R$ to denote the index set for *random samples*; that is, for $t \in \mathcal{W}_R$, the decision-maker randomly selects K products from the product candidate set as the assortment offered to the user arriving at time t. In §4.4, we detail the mechanics of how these random samples are generated via the random decay sampling schedule. Let n_T denote the size of the nonempty index set $\mathcal{W}_R$, i.e., $n_T = |\mathcal{W}_R| > 0$. The Lasso estimator for the unknown coefficient vector β^* can be defined as follows:

$$\hat{\beta} = \arg\min_{\beta} L(\beta) + \lambda \|\beta\|_1, \text{ where } L(\beta) := \frac{1}{n_T} \sum_{t \in \mathcal{W}_R} \log \left(e^{x_{c_t}^T \beta} / \left(1 + \sum_{i \in A_t} e^{x_i^T \beta} \right) \right). \tag{2}$$

The λ in Eq. 2 is a positive regularization parameter and decreases in the random sample size n_T.

Compared to the standard maximum likelihood estimator, the Lasso estimator in Eq. 2 introduces a ℓ_1 penalty term, $\lambda\|\beta\|_1$, to retain significant features with nonzero coefficients while pushing coefficients of insignificant features towards zero. Note that the Lasso estimator is identified in Eq. (2) merely by using *random samples* in the index set $\mathcal{W}_R$, but not by using *all samples* observed up to time T. This is because that these random samples preserve the iid property necessary for the desired asymptotic performance of the Lasso estimator.

To ensure the identifiability of the Lasso estimator, we need the following compatibility condition:

Assumption A.2 (Identifiability under the original space): There exists a $\kappa > 0$ such that for all vector u with $3\|u_S\|_1 \geq \|u_{S^c}\|_1$ and $|S| \leq s$, we have $\mathbb{E}\left[u^T \nabla^2 L(\xi) u\right] \geq \frac{\kappa}{s} \|u_S\|_1^2$, where ξ is any feasible solution.

The Assumption A.2 is a standard technical assumption in the Lasso literature (Candes et al. 2007, Bickel et al. 2009, Bühlmann and Van De Geer 2011) that regulates the covariance matrix's behavior in a restricted region and is necessary to ensure that the Lasso estimator asymptotically converges to its true value with high probability. We can show that the Lasso estimator defined in Eq. (2) satisfies the following inequality:

LEMMA 1. *Set the parameter* $\lambda = 2\sqrt{2x_{max}^2(\log d + \log T)/n_T}$. *Under Assumptions A.1-A.2, when* $n_T = \mathcal{O}(s^2 \log T)$, *there exists a constant* C_{lasso} *such that the event* $\mathcal{E}_{lasso}(T) := \left\{ \|\hat{\beta} - \beta^*\|_2 \leq C_{lasso} \cdot s \sqrt{\frac{\log d + \log T}{n_T}} := \mathcal{G}_0(T) \right\}$ *holds with probability* $1 - \mathcal{O}(1/T)$.

In a nutshell, Lemma 1 demonstrates that when the random sample size n_T is large enough, the Lasso estimator $\hat{\beta}$ will be close to the true feature coefficient β^* with high probability. Moreover, it is directly to show that as the random sample size n_T increases, $\mathcal{G}_0(T)$ decreases towards 0, which suggests that the Lasso estimator asymptotically converges to its true value.

Recall that through introducing the ℓ_1 penalty term, the Lasso method can perform feature selection by identifying significant features. In the following theorem, we show the effectiveness of the Lasso's feature selection capability under the MNL model.

THEOREM 1. *Denote* $\mathcal{S} = \{j : |\hat{\beta}_j| \geq 2\mathcal{G}_0(T)\}$ *and* $\beta_{\min} = \min_{j \in \mathcal{S}^*} |\beta_j^*|$. *Under the event* $\mathcal{E}_{lasso}(T)$, *the following three statements hold:*

1. $|\beta_j^*| \leq 3\mathcal{G}_0(T)$ *for all* $j \notin \mathcal{S}$.
2. *When* $\mathcal{G}_0(T) < \frac{1}{3}\beta_{\min}$, *we have* $\beta_j^* = 0$ *for all* $j \notin \mathcal{S}$.
3. $|\mathcal{S}| \leq s$.

The first statement of Theorem 1 demonstrates that when the feature j is not in the index set for significant features identified by the Lasso (i.e., $j \notin \mathcal{S}$), then its underlying true coefficient value β_j^* will be small. In other words, features outside of the index set $\mathcal{S}$ will have little influence on the user's utility and choice probabilities, regardless of whether this feature is actually a significant feature, $j \in \mathcal{S}^*$, or an insignificant feature, $j \notin \mathcal{S}^*$. Further, if the random sample size is sufficiently large so that $\mathcal{G}_0(T) < \frac{1}{3}\beta_{\min}$, or equivalently $n_T > (3C_{lasso}s/\beta_{min})^2(\log d + \log T)$, then the second statement shows that features outside of the index set $\mathcal{S}$ are indeed insignificant (i.e., if $j \notin \mathcal{S}$, then $j \notin \mathcal{S}^*$) and therefore have no influence on the user's utility and choice.

Hereafter, we informally refer to the scenario under which the random sample size is not large so that $n_T \leq (3C_{lasso}s/\beta_{min})^2(\log d + \log T)$ to be *the data-poor regime* and denote the scenario under which the random sample size is large enough so that $n_T > (3C_{lasso}s/\beta_{min})^2(\log d + \log T)$ as *the data-rich regime*. Below, in §4.2 and §4.4, we formally demonstrate that periodically using the Lasso to update the index set $\mathcal{S}$ will limit the influence of information loss, which is intrinsic and inevitable for random projection, and will ensure that the Lasso-RP-MNL algorithm attains the desired upper cumulative regret bound under both the data-poor and the data-rich regimes.

The last statement of Theorem 1 shows that the index set for significant features identified by the Lasso has a lower dimension than the dimension of true significant features ($|\mathcal{S}| \leq s = |\mathcal{S}^*|$). Therefore, parameter estimation for significant features can be performed in a lower dimension than its true underlying dimension to improve the computational efficiency.

4.2. Random Projection and Coefficient Estimation

If the decision-maker relies merely on features in the significant feature set identified by the Lasso to assess users' utilities and choices, then he can reduce the original high-dimensional parameter

estimation problem to a low-dimensional one by ignoring all features outside of the index set $\mathcal{S}$. Theorem 1 theoretically guarantees that in the data-rich regime, doing so may not significantly undermine the accuracy of parameter estimation and utility assessment.

In practice, however, the decision-maker may not always have the luxury of obtaining sufficient random samples, due to high costs (Bastani and Bayati 2015) or limited time (Wang et al. 2018). Under these scenarios, the Lasso may erroneously include insignificant features and exclude some significant features in the underlying true model, which causes the *model misspecification* problem. As many significant features/information will be hidden outside of the index set $\mathcal{S}$, ignoring these details will lead to a suboptimal assortment selection, lowering the decision-maker's reward. However, estimating coefficients for all features outside of the index set $\mathcal{S}$ will still be time-consuming, because these features remain high-dimensional. Therefore, to recycle information contained in these features, we propose reducing the dimensionality of these features to a low-dimensional space via random projection and then estimating coefficients for features in both the index set $\mathcal{S}$ and the projected low-dimensional space.

In a nutshell, random projection achieves dimension reduction by multiplying the original high-dimensional matrix by a random projection matrix, resulting a low-dimensional subspace with the same number of samples but fewer projected features. The Johnson-Lindenstrauss Lemma (Johnson and Lindenstrauss 1984) shows that the distance among points under the original high-dimensional space can be largely preserved under the projected low-dimensional space with high probability, and many theoretical studies and empirical applications have demonstrated the value of random projection as a computationally efficient method for dimension reduction (Pilanci and Wainwright 2015). In this study, we will project high-dimensional $(d-|\mathcal{S}|)$ features outside of the index set $\mathcal{S}$ into a low-dimensional m projected features by multiplying a random projection matrix $P \in \mathbb{R}^{m\times(d-|\mathcal{S}|)}$.

There are two popular choices for the random projection matrix P in the literature: Gaussian random projection matrix and sparse random projection matrix. In Gaussian random projection matrix, each entry $P_{i,j}$ is i.i.d. distributed and follows Gaussian distribution $N(0, 1/m)$, whereas in spare random projection matrix, entries take values $\{-\sqrt{v}, 0, \sqrt{v}\}$ with probabilities $\{1/(2v), 1 - 1/v, 1/(2v)\}$, where $v > 1$ is a parameter selected by the decision-maker. Clearly, increasing v decreases the number of nonzero elements in spare random projection matrix – the projection matrix becomes sparser. Therefore, spare random projection matrix is faster to generate, manipulate, and store than Gaussian random projection matrix. Yet, projecting high-dimensional data into a low-dimensional space will inevitably result in *information loss* (e.g., the Euclidean distance under the original high-dimensional space may not be precisely preserved under the projected low-dimensional space), so the cost of choosing sparse random projection is additional information loss

in preserving the pairwise distances (Li et al. 2006). In this research, we focus on Gaussian random projection matrix for a tighter theoretical regret bound.

First, we can bound the distance between the original high-dimensional vector and the projected low-dimensional vector with a certain probability guarantee.

LEMMA 2. *[Similar to Lemma 2.3.1 in Matoušek 2013] Let $P = (p_{ij})$ be a random $n \times m$ matrix such that each entry p_{ij} is chosen independently according to $N(0, 1/m)$. For any vector $u \in \mathbb{R}^n$ and $\epsilon > 0$, there exists a $C_2 > 0$ such that the event $\mathcal{E}_{rp}(m, n, \epsilon) := \{|\|Pu\| - \|u\|| \leq \epsilon \|u\|\}$ holds for all $\epsilon \in (0, 1)$ with probability $1 - 2\exp(-4C_2\epsilon^2 m)$.*

Lemma 2 demonstrates that Gaussian random projection can largely preserve the geometry structure of the original vector with reasonable distortions with high probability. Hence, by adopting random projection techniques, the decision-maker can significantly reduce the computational time without much sacrifice to the accuracy of parameter estimation. Yet, distortions or information loss in the process of projecting high-dimensional data to a low-dimensional space could lead to a worse regret performance, because such distortions do not vanish over time (Kuzborskij et al. 2018). To limit the negative influence of information loss in random projection, we propose combining random projection with the Lasso.

Recall that as the random sample size increases, the Lasso can learn and gradually identify significant features in the index set $\mathcal{S}$ (Theorem 1). When the random sample size is large enough, most useful information will already be contained within these features identified by the Lasso. Therefore, the Lasso-RP-MNL algorithm will only project high-dimensional features *outside* of the index set $\mathcal{S}$ to a low-dimensional space and then estimate coefficients for features in both the index set $\mathcal{S}$ and the projected space. Therefore, the long-term information loss (due to random projection) will be limited by the Lasso, and the negative influence of model misspecification (due to the Lasso) can be mitigated by recycling features outside of the index set $\mathcal{S}$ via random projection.

Now, given an index set $\mathcal{S}$, we describe the process of constructing the projection matrix P_0 and the permutation matrix Q. Through these two matrices, the decision-maker can keep features in the index set $\mathcal{S}$ unchanged while randomly projecting the remaining $(d - |\mathcal{S}|)$ features to a lower m dimensions. To this end, we first need to generate a random projection matrix $P \in \mathbb{R}^{m \times (d - |\mathcal{S}|)}$, where $P_{i,j}$ follows Gaussian distribution $N(0, 1/m)$. Then, combining the random projection matrix P with an identity matrix $I \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{S}|}$, we can construct the projection matrix $P_0 = \begin{pmatrix} I & 0 \\ 0 & P \end{pmatrix} \in \mathbb{R}^{(|\mathcal{S}|+m) \times d}$. Next, we use $Q \in \mathbb{R}^{d \times d}$ to denote the permutation matrix, which moves the original feature vector x's significant features in the index set $\mathcal{S}$ to the top $|\mathcal{S}|$ places in the permuted feature vector Qx. Hence, by multiplying the projection matrix P_0 by the permuted feature vector Qx, we project the original d dimensional vector to a low-dimensional $(|\mathcal{S}| + m)$ vector, in which the

first $|\mathcal{S}|$ elements are original features in the index set $\mathcal{S}$ identified by the Lasso and the remaining m elements are the projected features by projecting the original features not in the index set $\mathcal{S}$ via the random projection matrix P. For simplicity of notation, we define z as the projected feature vector, $z := P_0 Q x$. Similarly, the following notations are used throughout this paper: $\theta^* := P_0 Q \beta^*$ and $\Sigma := Q^T P_0^T P_0 Q$.

THEOREM 2. *Let matrix P_0 and Q be constructed by the index set $\mathcal{S} := \{j : |\hat{\beta}_j| \geq 2\mathcal{G}_0(T)\}$. When event $\mathcal{E}_{lasso}(T)$ holds, there exist positive constants C_1 and C_2 such that the event $\mathcal{E}_2(m,T) := \left\{\|\beta^* - \Sigma\beta^*\| \leq \frac{s(C_1^2+\log d)+(1/C_1+1/C_2)\log T+m}{m} \cdot 6\sqrt{s}\mathcal{G}_0(T) := \mathcal{G}_1(m,T)\right\}$ holds with probability $1-\mathcal{O}(1/T)$ for $m \leq \log T/C_1 + s\log d$. Furthermore, when $\mathcal{G}_0(T) < \frac{1}{3}\beta_{min}$, we have $\|\beta^* - \Sigma\beta^*\| = 0$.*

Theorem 2 demonstrates that the true feature coefficient vector β^* is nearly invariant under the projection Σ, which directly suggests that our proposed projection scheme is nearly optimal in the sense that it will not introduce estimation error when predicting users' utilities and choices asymptotically. Specifically, consider projecting both the feature vector x and the coefficient vector β^* by using P_0 and Q. Then, the projected utility can be written as $(P_0 Q x)^T P_0 Q \beta^* = x^T \Sigma \beta^*$. Note that as illustrated in Theorem 2, the time dependence of the term $\|(I-\Sigma)\beta^*\|$ is on the order of $\mathcal{G}_0(T)\log T \approx n_T^{-1/2}\log^{3/2} T$. Therefore, if we can ensure that the random sample size n_T is on the order of at least $\mathcal{O}(T^c)$ up to time T (to be detailed in §4.4), where c is an arbitrary positive constant, then $\|(I-\Sigma)\beta^*\|$ will converge to 0 with high probability.

By combining the Lasso and random projection, the decision-maker can project the original high-dimensional d features into a low-dimensional $(|\mathcal{S}|+m)$ space so that the parameter estimation can be performed in a low-dimensional fashion. Specifically, we estimate the coefficients for the projected feature vector $z = P_0 Q x$ as follows:

$$\hat{\theta} = \arg\min_{\|\theta-\theta_0\|\leq\tau} L_z(\theta), \text{ where } L_z(\theta) := \frac{1}{T}\sum_{t=1}^{T} \log\left(e^{z_{c_t}^T\theta} / \left(1 + \sum_{i\in A_t} e^{z_i^T\theta}\right)\right). \tag{3}$$

The τ in Eq. 3 is a positive constant selected by the decision-maker and $\theta_0 = \arg\min_\theta \|\theta - P_0 Q\hat{\beta}\|$. The $\|\theta-\theta_0\| \leq \tau$ is a local constraint added in Eq. (3) to prevent over-fitting, and we solve $\hat{\theta}$ only in the local space around θ_0. Because from Lemma 1, we know that $\hat{\boldsymbol{\beta}}$ will not be far away from $\boldsymbol{\beta}^*$, it implies that $\hat{\theta}$ is also close to $P_0 Q\beta^*$ with high probability.

As the projected feature vector $P_0 Q x$ is low-dimensional with $(|\mathcal{S}|+m)$ dimensions, solving Eq. (3) is more computationally efficient than solving Eq. (2), which is a high-dimensional estimation problem with the original d-dimension features. Another difference between these two equations is that estimating coefficients for the projected feature vector in Eq. (3) uses all available samples, whereas the Lasso estimator in Eq. (2) relies merely on random samples in the index set $\mathcal{W}_R$.

Similarly to Assumption A.2, which ensures the identifiability of the Lasso estimator in Eq. (2) under the original high-dimensional space, we need the last technical assumption, which requires L_z to be strongly convex under the projected space, to achieve the identifiability of the estimator $\hat{\theta}$ in Eq. (3) under the projected space:

Assumption A.3 (Identifiability under the projected space): When all samples in $L_z(\theta)$ are i.i.d. random samples, there exists a $\mu > 0$ such that for any v and feasible solution ξ in the projected space, we have $\mathbb{E}\left[v^T \nabla^2 L_z(\xi) v\right] \geq \mu \|v\|^2$.

4.3. Assortment Selection

In this subsection, using the estimated coefficient vector in the projected space, we construct the upper-confidence bound for each individual product's attraction parameter, identify the optimistic assortment, and establishes the single-period regret upper bound for the optimistic assortment.

Given an arbitrary assortment $\mathcal{A}$, we denote the decision-maker's expected reward for a coefficient vector θ under the projected space as

$$\mathcal{R}_{\mathcal{A}}(\theta) = \sum_{i \in \mathcal{A}} \frac{r_i e^{z_i^T \theta}}{1 + \sum_{j \in \mathcal{A}} e^{z_j^T \theta}}.$$

The following Lemma establishes an upper bound on the expected reward difference between the estimator $\hat{\theta}$ in Eq. (3) under the projected space and the projected true coefficient vector $P_0 Q \beta^*$:

LEMMA 3. *Denote* $f_{\mathcal{A}}(\theta) = \mathbb{E}[p_{\Sigma\beta^*,\mathcal{A}}(i) \log(p_{Q^T P_0^T \theta,\mathcal{A}}(i)/p_{\Sigma\beta^*,\mathcal{A}}(i))]$ *and* $\delta = \|\hat{\theta} - P_0 Q \beta^*\|$. *Let* L_3 *and* λ_{max} *be positive constants such that for all* $t > 0$ *and feasible* θ_2, θ_1, ξ, *we have* $\|\nabla^2 f_{\mathcal{A}_t}(\theta_1) - \nabla^2 f_{\mathcal{A}_t}(\theta_2)\|_{op} \leq L_3 \|\theta_1 - \theta_2\|$ *and* $\|\nabla^2 \mathcal{R}_{\mathcal{A}_t}(\xi)\|_{op} \leq \lambda_{\max}$. *Under Assumption A.1 and A.3, if* $\delta \leq \min\{\frac{n_T \mu}{4TL_3}, \frac{\rho}{8Kx_{\max}}\}$, $\mathcal{G}_1(m,T) \leq \frac{\rho}{8Kx_{\max}}$, *and events* $\mathcal{E}_2(m,T)$ *and* $\mathcal{E}_{rp}(m,d,1/2)$ *hold, then the following inequality holds for all assortment* $\mathcal{A}$ *with probability* $1 - \mathcal{O}(1/T)$*:*

$$|\mathcal{R}_{\mathcal{A}}(\hat{\theta}) - \mathcal{R}_{\mathcal{A}}(P_0 Q \beta^*)| \leq \sqrt{2} R_{\max} \omega_T \sqrt{\left\| \left(\sum_{i=1}^{T} \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-\frac{1}{2}} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) \left(\sum_{i=1}^{T} \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-\frac{1}{2}} \right\|_{op}} + \frac{1}{2} \lambda_{\max} \delta^2, \tag{4}$$

where $\omega_T = 4\sqrt{4x_{\max}^2 T \mathcal{G}_1(m,T)\delta + 32(C_3(s+m)+1)\log(T) + 2\Gamma_T}$, $\Gamma_T = \max\{0, TL_z(\hat{\theta})\}$, *and* C_3 *is a positive constant.*

The upper bound established in Lemma 3 has two components. The first term in the right-hand-side of Eq. (4) is the typical UCB-type upper-confidence bound, whereas the additional second term, $\lambda_{\max}\delta^2/2$, accounts for the possible influence of model misspecification and information loss. In the classic setting, we assume that the true model always falls in the solution space so that the estimator enjoys asymptotical unbiasedness. However, this assumption no longer holds

when the Lasso fails to identify all significant features and random projection is used to compress all remaining high-dimensional features, potentially including true significant features. In such a scenario, the decision-maker will face an upper bound worse than the typical UCB-type bound.

Yet, such negative influences of model misspecification and information loss should not be of much concern under the data-rich region. In particular, note that in Lemma 3, we require that δ converges to zero at the rate of $\mathcal{O}(n_T/T)$, and therefore the additional second term $\lambda_{\max}\delta^2/2$ diminishes at the rate of $\mathcal{O}(T^{-1/2})$, if we ensure $n_T = \tilde{\mathcal{O}}(T^{1/2})$ (see §4.4 for details). Hence, under the data-rich region, the upper bound established in Lemma 3 converges to the typical UCB-type upper-confidence bound.

When the given assortment includes merely a single item, we can establish an upper bound for each individual product's attraction parameter. Specifically, consider a single-item assortment $\mathcal{A}$ that contains a single product with the feature vector x and reward $r = 1$. The decision-maker's expected reward under the projected space can be simplified to $(e^{x^T\beta^*}/(1+e^{x^T\beta^*}))$, and the attraction parameter can be upper bounded as in the following corollary.

Corollary 1. *Let $\mathcal{A}$ be the assortment with a single item characterized by the feature vector x. If the same conditions stated in Lemma 3 hold, then with probability $1-\mathcal{O}(1/T)$, we have $e^{x^T\beta^*} \le v^{ucb}$, where*

$$\begin{aligned} v^{ucb} =& e^{x^T Q^T P_0^T \hat{\theta}} + e^{x_{\max} b} x_{\max} \mathcal{G}_1(m,T) + \frac{\lambda_{\max}}{2\phi'(e^{x_{\max} b})}\delta^2 \\ &+ \frac{\sqrt{2}R_{\max}\omega_t}{\phi'(e^{x_{\max} b})} \sqrt{\left\| \left(\sum_{i=1}^{T} \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-1/2} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) \left(\sum_{i=1}^{T} \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-1/2} \right\|_{op}} \end{aligned} \tag{5}$$

and $\phi(x) = x/(1+x)$.

This upper bound for the single product' attraction parameter will facilitate the analysis of the regret upper bound for the decision-maker's "optimal" static assortment. In particular, given the current estimator $\hat{\theta}$, in order to maximize his single-period expected reward, the decision-maker needs to offer at most K products out of N candidates. Equivalently, the decision-maker solves the following static assortment optimization problem under the capacity constraint:

$$\max_{\substack{\mathcal{A}\subseteq\{1,2,\ldots,N\} \\ |\mathcal{A}|\le K}} \left\{ R_{\mathcal{A}}\left(\hat{\theta}\right) \right\}.$$

The key challenge in identifying the optimal assortment is that the decision-maker must search through a combinatorial space of N products. In practice, the number of products can easily exceed hundreds or thousands, which makes the problem computationally intractable for online assortment

optimization problems. Therefore, following Davis et al. (2013), we reformulate this combinatorial optimization problem as a linear programming problem:

$$\max_{\boldsymbol{w}} \sum_{i\in N} r_i w_i, \text{ s.t. } \sum_{i\in N} w_i + w_0 = 1;\ \sum_{i\in N} \frac{w_i}{v_i} \le K w_0;\ 0 \le w_i \le w_0 v_i \text{ for } i \in N. \tag{6}$$

As at most K decision variables will be none-zero under the optimal solution, various efficient solution algorithms, such as column-generation techniques, can be adopted to expedite the computation for this LP problem. Now, we replace the product i's attraction parameter v_i in Eq. (6) by v_i^{ucb} and denote the optimal solution to the resulting problem as $\boldsymbol{w}^*$. Then, we refer to the assortment $\mathcal{A}^{SRP} = \{i \in N : w_i^* > 0\}$ as the static assortment under random projection. It is also worth noting that that the static assortment under random projection $\mathcal{A}^{SRP}$ may not be the true optimal assortment under the original high-dimensional space for two reasons: The projected space may not contain the true coefficients, where the systemic bias may appear, and the estimator $\hat{\theta}$ may not match the best possible candidate of the true coefficients in the projected space.

Now, we denote the decision-maker's expected reward for a given assortment $\mathcal{A}$ under the true coefficient β^* in the original high-dimensional space as

$$\mathcal{R}_{\beta^*}(\mathcal{A}) = \sum_{i\in\mathcal{A}} \frac{r_i e^{x_i^T \beta^*}}{1 + \sum_{j\in\mathcal{A}} e^{x_j^T \beta^*}}.$$

Further, we use $\mathcal{A}^*$ to denote the optimal assortment, which can be identified by searching the combinatorial space of all products to maximize $\mathcal{R}_{\beta^*}(\mathcal{A})$, i.e., $\mathcal{A}^* = \arg\max_{\mathcal{A},|\mathcal{A}|\le K} \mathcal{R}_{\beta^*}(\mathcal{A})$. Next, we bound the expected reward difference between the static assortment under random projection $\mathcal{A}^{SRP}$ and the optimal assortment $\mathcal{A}^*$ in the following theorem:

THEOREM 3. *Let $\mathcal{A}^{SRP}$ be the static assortment under random projection and $\mathcal{A}^*$ be the optimal assortment. Under the same conditions as in Lemma 3, the following inequality holds with probability $1 - \mathcal{O}(1/T)$:*

$$\mathcal{R}_{\beta^*}(\mathcal{A}^*) - \mathcal{R}_{\beta^*}(\mathcal{A}^{SRP}) \le R_{\max} K \eta x_{\max}(2\delta + 2\mathcal{G}_1(m,T)) + \frac{R_{\max} K \lambda_{\max}}{2\phi'(\eta)}\delta^2$$
$$+ \frac{R_{\max}^2 K \eta^{3/2}(1+K\eta)\sqrt{2}\omega_t}{\phi'(\eta)(1+\eta)} \sqrt{\left\| \left(\sum_{i=1}^{T} \nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2} \nabla^2 f_{\mathcal{A}^{SRP}}(\hat{\theta}) \left(\sum_{i=1}^{T} \nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2} \right\|_{op}},$$

where $\eta = \exp(x_{\max} b)$.

The upper regret bound for the static assortment under random projection $\mathcal{A}^{SRP}$ can be divided into two parts. On the one hand, the first part, $R_{\max} K \eta x_{\max}(2\delta + 2\mathcal{G}_1(m,T)) + \frac{R_{\max} K \lambda_{\max}}{2\phi'(e^{x_{\max} b})}\delta^2$, comes from the single product utility decomposition. The magnitude of the first part of the upper-confidence bound can be regulated by a judiciously designed random sample schedule. In particular, in the data-poor regime, it is straightforward to show that $\mathcal{G}_1(m,T) = \mathcal{O}(\sqrt{\log T / n_T})$ and

$\delta = \mathcal{O}(n_T/T)$. Therefore, if we can design the random sample size n_T to be on the order of at least $\tilde{\mathcal{O}}(T^{2/3})$, then the first part will be upper-bounded at most by the order $\tilde{\mathcal{O}}(T^{-1/3})$. When switching to the data-rich regime (i.e., $\mathcal{G}_0(T) \leq \frac{1}{3}\beta_{\min}$), where we have $\mathcal{G}_1(m,T) = 0$ with high probability (Theorem 2). If we design $n_T = \tilde{\mathcal{O}}(T^{1/2})$ random sampling scheduling, the first part of the upper-confidence bound will vanish at a faster $\tilde{\mathcal{O}}(T^{-1/2})$ rate. On the other hand, the second part is the typical UCB-type upper-confidence bound, which shares the typical quadratic upper bound as in UCB-type algorithms and can be further bounded by the elliptical potential lemma (see Dani et al. 2008, Rusmevichientong et al. 2010, Filippi et al. 2010, Li et al. 2017).

It is worth mentioning that the common K parameter in the right-hand-side stems from the fact that we construct the product-based bound instead of the assortment-based bound. In particular, instead of enumerating all possible combinations (choose K out of N products) and building confidence bounds for each combination (as in Chen et al. 2018), we construct confidence bounds for each product (see Corollary 1) so that the estimation error will accumulate with respect to the assortment size K. Although constructing the product dependent bound instead of assortment dependent bound results in additional cost, the decision-maker benefits from the significant reduction in computational time. For example, consider a scenario where the decision-maker selects 20 out of $1,000$ products: Instead of constructing 3.4×10^{41} confidence bounds for assortments, we need to build only 10^3 confidence bounds – one confidence bound for each product. Hence, in practice, where the decision-maker typically faces a large product candidate set, our algorithm becomes more efficient and pragmatic in online decision-making settings.

4.4. Lasso-RP-MNL Algorithm

Recall that as the number of random samples increases, the negative influences of model misspecification in the Lasso and information loss in random projection can be efficiently mitigated (Theorem 1 and 2). Further, note that the regret upper bounds on selected assortments are contingent on the number of random samples (see related discussions for Lemma 3 and Theorem 3). Therefore, before presenting the Lasso-RP-MNL algorithm, we first propose the random decay sampling schedule that generates sufficient, but not excessive to compromise the decision-maker's reward performance, random samples.

***Random Decay Sampling Schedule*:** At the beginning of each time t, the decision-maker draws a random number r_t that follows Bernoulli distribution with success probability

$$\mathbb{P}(r_t = 1) = \min\left\{1, C_0\left[\frac{1}{t^{\frac{1}{2}}} + \frac{\mathbb{1}(\mathcal{G}_0(t) \geq \frac{1}{3}\beta_{\min})}{t^{\frac{1}{3}}}\right]\right\} := P_{C_0}(t),$$

where C_0 is selected by the decision-maker. If $r_t = 1$, then the decision-maker randomly selects K products from the candidate set as the assortment to the user arriving at time t.

Note that under the random decay sampling schedule, the random sampling probability $P_{C_0}(t)$ first decreases at a lower rate of $t^{-\frac{1}{3}}$ and then at a higher rate of $t^{-\frac{1}{2}}$. Therefore, we can deliberately select a C_0 value so that the total number of random samples generated will be on the order of $\mathcal{O}(C_0 T^{\frac{2}{3}})$ in the data-poor regime and then on the order of $\mathcal{O}(C_0 T^{\frac{1}{2}})$ in the data-rich regime. The high sampling rate at the beginning helps generate sufficient random samples so that the index set $\mathcal{S}$ can be updated more frequently via the Lasso to hedge against model misspecification. In the data-rich regime, when model misspecification has been contained, the conventional $\mathcal{O}(C_0 T^{\frac{1}{2}})$ random sampling rate can be deployed to maintain the feature selection accuracy asymptotically.

The proposed Lasso-RP-MNL algorithm can be presented as follows:

Algorithm : Lasso-RP-MNL Algorithm

Require: Input C_0, m, λ_0, and the Lasso step set T_{lasso}. Initialize $t = 1$, $\mathcal{W}_R = \emptyset$, $\mathcal{W} = \emptyset$, $P_0 \in \mathbb{R}^{m \times d}$ with i.i.d $N(0, 1/m)$ Gaussian random elements, $Q = I$, $\theta_0 = \mathbf{0}$, $\{\omega_t\}$, and $\tau = +\infty$.

for $t = 1, 2, ...$ **do**

Draw a Bernoulli random number b_t with success probability $P_{C_0}(t)$.

if $b_t = 1$ **then**

1. Randomly select K products from the candidate set $\{1, 2, ..., N_t\}$ as the assortment $\mathcal{A}_t$.

2. Observe the user's choice $c_t \in \mathcal{A}_t \cup \{0\}$ and update $\mathcal{W}_R = \mathcal{W}_R \cup \{t\}$ and $\mathcal{W} = \mathcal{W} \cup \{t\}$.

else

1. Solve Eq. (3) for $\hat{\theta}$ with samples in $\mathcal{W}$ and update attraction parameters' upper bounds $v_i = v_i^{ucb}$ for $i \in \{1, 2, ..., N_t\}$ according to Eq. (5).

2. Plug the updated v_i back to Eq. (6), solve for $\boldsymbol{w}^*$, and offer $\mathcal{A}_t = \{i \in N_t : w_i^* > 0\}$.

3. Observe the user's choice $c_t \in \mathcal{A}_t \cup \{0\}$ and update $\mathcal{W} = \mathcal{W} \cup \{t\}$.

end if

if $t \in T_{lasso}$ **then**

1. Solve Eq. (2) for $\hat{\beta}$ with samples in $\mathcal{W}_R$ and $\lambda = \lambda_0 \sqrt{(\log d + \log t)/|\mathcal{W}_R|}$, and update the index set for significant features $\mathcal{S} = \{j : |\hat{\beta}_j| \geq 2\mathcal{G}_0(t)\}$.

2. Re-construct the projection matrix $P_0 = \begin{pmatrix} I & 0 \\ 0 & P \end{pmatrix} \in \mathbb{R}^{(|\mathcal{S}|+m) \times d}$, where $P \in \mathbb{R}^{m \times (d - |S|)}$ with i.i.d $N(0, 1/m)$ random elements, update the permutation matrix $Q \in \mathbb{R}^{d \times d}$ so that features in $\mathcal{S}$ are moved to the top $|\mathcal{S}|$ places in Qx, and set $\theta_0 = \arg\min \|\theta - P_0 Q \hat{\beta}\|$ and $\tau = \min\{\frac{\rho}{8Kx_{\max}}, \frac{n_t \mu}{4tL_3}\}$.

end if

end for

The Lasso-RP-MNL algorithm starts with assigning values for system parameters and initialing intermediate matrices and variables. For a user arriving at time t, the decision-maker will follow the random decay sampling schedule to draw a Bernoulli random number. If this random number equals 1, then the decision-maker will randomly select K products from the product candidate set, offer the resulting assortment to the user, observe the user's choice, and finally include this sample in both the random sample index set $\mathcal{W}_R$ and the whole sample index set $\mathcal{W}$.

Otherwise, if the Bernoulli random number equals 0, then the decision-maker will first estimate the coefficients for the projected low-dimensional feature vector $\hat{\theta}$, based on which the decision-maker will update the attraction parameters' upper-confidence bounds for all products in the candidate set. Next, the decision maker will treat these upper-confidence bounds as new attraction parameters in the MNL model and plug them back into Eq. (6) to identify the assortment that maximizes his expected reward. Then, the decision-maker will offer this assortment to the user, observe the user's choice, and include this sample in the whole sample index set $\mathcal{W}$ only.

Finally, before moving to the next user, the decision-maker will check whether the current time t belongs to a predetermined Lasso step set T_{lasso}. If not, then the decision-maker does not need to do anything and will move directly to the next user. Otherwise, if $t \in T_{lasso}$, then the decision-maker will update the index set for significant features $\mathcal{S}$ via the Lasso using only random samples in the index set $\mathcal{W}_R$, reconstruct the projection matrix P_0 and the permutation matrix Q, recalculate the local solution θ_0, and then move to the next user.

Note that the Lasso-RP-MNL algorithm runs the Lasso to update the index set $\mathcal{S}$ only when the current time t belongs to the set T_{lasso}. This is because solving the Lasso problem can be time-consuming, which makes it impractical to update the Lasso for every arriving user under the online decision-making scenario. Hence, to tackle this computational challenge, we construct a very sparse Lasso step set such that the number of users between two consecutive Lasso runs increases exponentially. In particular, we set $T_{lasso} = \{t : t = c^i,\ i = 0, 1, 2, ...\}$ with a positive integer $c > 1$. Therefore, as time progresses, the frequency of updating the Lasso decreases at an exponential rate, which alleviates the computational burden associated with solving the Lasso under high-dimensional data with large sample sizes, while maintaining proper accuracy for parameter estimation.

Further, in practice, the Lasso updates and the assortment optimizations can be computed in parallel to further trim down the computational time. Specifically, if the current time is selected to update the Lasso, then the decision-maker does not need to wait for the Lasso run to conclude before moving to the next user; instead, while the Lasso is updating the new index set $\mathcal{S}$, this decision-maker can rely on previously constructed projection matrix P_0, permutation matrix Q, and local solution θ_0 to select assortments for incoming users.

We are now ready to establish the expected cumulative regret upper bound for the Lasso-RP-MNL algorithm.

THEOREM 4. *Under Assumptions $A1-A3$, if we set $C_0=\mathcal{O}(s^2\log d\log T)$ for the random decay sampling schedule, $T_{lasso}=\{c^i,\ i=0,1,2,...\}$ with a positive integer $c>1$ for the Lasso step set, $\lambda_0=2\sqrt{2}x_{\max}$, $m\leq \log T/C_1+s\log d$, and $\frac{R_{\max}^2K\eta^{3/2}(1+K\eta)\sqrt{2}\omega_t}{\phi'(e^{x_{\max}b})(1+\eta)}\geq 1$, then there exists a $T_0>0$ such that for all $T\geq T_0$, with probability $1-2\exp(-C_2m)-\mathcal{O}(T^{-1})$, the expected cumulative regret of the Lasso-RP-MNL algorithm is upper bounded by*

$$\textit{Regret}(T)\lesssim\begin{cases}\mathcal{O}\left(s^{\frac{11}{4}}m^{\frac{3}{2}}\cdot\log d\cdot T^{\frac{2}{3}}\log^4 T\right)\ , \mathcal{G}_0(T)\geq\frac{1}{3}\beta_{\min};\\ \mathcal{O}\left(s^2m\cdot\log d\cdot T^{\frac{1}{2}}\log^3 T\right)\quad , \mathcal{G}_0(T)<\frac{1}{3}\beta_{\min}.\end{cases}$$

Theorem 4 demonstrates that for the feature dimension d and the sample size dimension T, the expected cumulative regret of the Lasso-RP-MNL algorithm is upper-bounded by $\tilde{\mathcal{O}}(\sqrt{T}\log d)$ in the data-rich regime. Compared to $\tilde{\mathcal{O}}(d\sqrt{T})$ for UCB-type bounds in Chen et al. (2018) and Oh and Iyengar (2019a), the Lasso-RP-MNL algorithm improves such a linear dependence on the feature dimension d to a logarithmic dependence. Further, in the absence of contextual information, Agrawal et al. (2017, 2019) adopt UCB and Thompson sampling techniques to establish the regret upper bound $\tilde{\mathcal{O}}(\sqrt{NT})$ for $T\gg N^2$. Therefore, when the logarithm of the feature dimension is smaller than the square root of the number of available products (i.e., $\log d<\sqrt{N}$), we can also improve their regret bound.

More importantly, we also show that in the data-poor regime with limited samples, the Lasso-RP-MNL algorithm is upper-bounded by $\tilde{\mathcal{O}}(T^{\frac{2}{3}}\log d)$. We want to highlight this result by comparing it to the scenario where the decision-maker relies only on the Lasso to perform dimension reduction. In particular, consider an auxiliary Lasso-only algorithm in which, keeping everything else unchanged, the decision-maker estimates coefficients only for significant features identified by the Lasso and ignores all remaining features. Under such an auxiliary Lasso-only algorithm, if the Lasso fails to fully identify significant features in the data-poor regime, which is highly possible due to insufficient random samples, then the decision-maker will rely on the misspecified model to perform coefficient estimation and assortment selection. Under this scenario, the expected single-step regret will be proportional to the strength of the model bias, leading to a linear cumulative regret $\tilde{\mathcal{O}}(T)$.

The Lasso-RP-MNL algorithm, however, estimates coefficients for two sets of features. The first set is the $|\mathcal{S}|$ significant features identified by the Lasso, and the second set is all remaining $(d-|\mathcal{S}|)$ features projected down to m dimensions by random projection. Therefore, the negative influence of model misspecification can be partially mitigated by recycling features in the second set, so we can improve the cumulative regret's dependence on time T from linear to sublinear, $\tilde{\mathcal{O}}(T^{\frac{2}{3}})$.

5. Empirical Experiments

In this section, we benchmark the Lasso-RP-MNL algorithm to existing state-of-the-art algorithms in the literature and industrial practices. In §5.1, we first explore the benefits of the Lasso-RP-MNL algorithm by comparing it to four benchmarks (i.e., MNL-Bandit by Agrawal et al. 2019 and other three ancillary algorithms) to illustrate the value of the high-dimensional contextual information and the value of dimension reduction techniques (i.e., the Lasso and random projection). Next, in §5.2, we simulate the real practice environment, where users are heterogeneous, the product candidate set is large, and the feature vector is high-dimensional, to examine the impacts of the size of the product candidate set N, the feature dimension d, and the projection dimension m on Lasso-RP-MNL's cumulative regret performance and computational time. Finally in §5.3, we use the high-dimensional XianYu online assortment recommendation dataset to evaluate the Lasso-RP-MNL algorithm's performance in a real practice scenario.

5.1. The Benefits of Lasso-RP-MNL: A Preliminary Illustration

The benefits of the Lasso-RP-MNL algorithm might be justified mainly by two key factors: incorporating high-dimensional contextual information and combining two dimension reduction techniques (i.e., the Lasso and random projection). Hence, to separately gauge the impacts of these two factors, we first introduce four benchmark algorithms in the first synthetic experiment, as follows:

- *MNL-Bandit*: Proposed by Agrawal et al. (2019), MNL-Bandit is a UCB-based algorithm without the contextual information.
- *Benchmark 1 (With Features)*: Benchmark 1 follows the same structure as the Lasso-RP-MNL algorithm, but without using the Lasso and random projection; this benchmark estimates the unknown coefficient vector β^* under the original high-dimensional space.
- *Benchmark 2 (RP Only)*: Benchmark 2 follows the same structure as the Lasso-RP-MNL algorithm, but it does not update the index set for significant features via the Lasso (i.e., $\mathcal{S} \equiv \emptyset$); instead, it projects all features into a low m-dimensional space via random projection.
- *Benchmark 3 (Lasso Only)*: Benchmark 3 follows the same structure as the Lasso-RP-MNL algorithm, but it estimates coefficients only for features in the index set $\mathcal{S}$ identified and periodically updated by the Lasso and ignores all remaining features outside of $\mathcal{S}$.

Note that MNL-Bandit does not use high-dimensional contextual information to estimate each user-product pair's utility. Instead, it assigns a unique attraction parameter for each product and directly estimates these attraction parameters using the sample average (see Agrawal et al. 2019). Therefore, comparing Benchmark 1 to MNL-Bandit[1] could shed light on the value of incorporating

[1] Besides Benchmark 1, MLE-UCB by Chen et al. (2018) can also be used to illustrate the value of contextual information. Yet, as MLE-UCB does not use dimension reduction techniques for learning and relies on computing individual assortment bounds to identify the optimal assortment, it is highly computational expensive and therefore is unsuitable for online assortment optimization under high-dimensional data.

the contextual information. Further, because it uses no dimension reduction technique for high-dimensional data, Benchmark 1 is not computationally efficient and will perform poorly under limited samples due to high variances and poor estimates. Therefore, by comparing Benchmark 2 and Benchmark 3 to Benchmark 1, we can separately assess the benefits of the Lasso and random projection under the high-dimensional online assortment optimization setting. Finally, we compare the Lasso-RP-MNL algorithm to Benchmark 2 and Benchmark 3 to gauge the benefits of combining the Lasso and random projection.

5.1.1. Data Generation and Parameter Inputs: In the first experiment, we consider that a decision-maker needs to offer at most 5 products (i.e., $K = 5$) out of a candidate set of 20 products (i.e., $N = 20$) to users. The number of unique features that identify these user-product pairs is set to be 20 (i.e., $d = 20$), and the feature vectors, x_i for $i = \{1, 2, ..., N\}$, are independently and identically generated from the standard Gaussian distribution.

Note that MNL-Bandit can not be directly applied to our setting with heterogeneous users and changing product candidate sets. This is because that the learning in MNL-Bandit is associated with the attraction parameter for each product. Therefore, to be able to learn these attraction parameters, MNL-Bandit requires that the number of products is not too large and the true values of these attraction parameters remain unchanged in the experiment. Yet, changing product candidate sets from user to user will significantly increase the number of products' attraction parameters, and including the influences of heterogeneous users (i.e., different user-feature values) to products will directly change their attraction parameters. Hence, to benchmark against MNL-Bandit, we will consider a setting where feature vectors x_i for $i = \{1, 2, ..., N\}$ contain only product features and the same group of feature vectors is repetitively offered to the decision-maker. In other words, in the first experiment, the decision-maker will repetitively choose from a fix set of 20 products to a group of T homogeneous users. Technically, we will generate feature vectors for 20 products once at the beginning of the experiment and offer these 20 products repetitively to the decision-maker for every incoming user. This constrain will be relaxed in other synthetic experiments in §5.2 and the XianYu online assortment recommendation experiment in §5.3.

The unknown true coefficient vector β^* is sparse, and only 5 out of a total of 20 coefficients are non-zero (i.e., $d = 20$ and $s = 5$). Without any loss of generality, we set the first five features to be significant (i.e., $\boldsymbol{\beta}^* = \{\beta_1^*, \beta_2^*, \beta_3^*, \beta_4^*, \beta_5^*, 0, 0, ..., 0\}$), and their values are also independently and identically generated by a Gaussian distribution. Finally, the corresponding reward r_i for $i = \{1, 2, ..., 20\}$ is generated from a uniform distribution from 0 to 1. In the experiment, we arbitrarily set the parameters $\lambda_0 = 1$, $C_0 = 3$, $c = 2$ and the projection dimension $m = 3$.

5.1.2. Results: For each algorithm, we perform 100 trials and report the average cumulative regret in Figure 1 for the first $1,000$ users (i.e., $T = 1,000$), at which time all algorithms, except Benchmark 2, seem to have converged. The average computational time (in seconds) for one trial is 11 for MNL-Bandit, 53 for Benchmark 1, 26 for Benchmark 2, 25 for Benchmark 3, and 33 for Lasso-RP-MNL. Recall that MNL-Bandit uses the sample mean to update its estimators for unknown attraction parameters instead of adopting MLE in all other algorithms. Therefore, MNL-Bandit tends to have better computational time performance, especially when the number of products in the candidate set N is not large. Without using any dimension reduction techniques, Benchmark 1 requires the longest computational time among all algorithms.

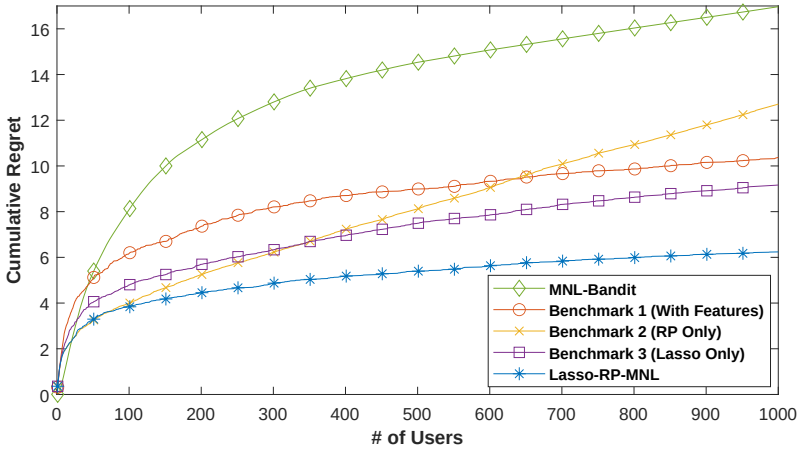


Figure 1 **The impact of T on the cumulative regret, where $d = 20$, $s = 5$, $N = 20$, $K = 5$, and $m = 3$.**

Figure 1 illustrates the cumulative regret performance for all algorithms. First, when comparing Benchmark 1 to MNL-Bandit, we observe that algorithms using available contextual information could significantly improve the decision-maker's regret performance. Second, note that as the underlying data possess a sparse structure, the decision-maker could use the Lasso (in Benchmark 3) to perform feature selection, identify the underlying sparse data structure, and improve the accuracy of parameter estimation. Indeed, we observe that with the Lasso, Benchmark 3 further reduces the decision-maker's cumulative regret from Benchmark 1. Third, the Lasso may suffer from model misspecification, especially in the data-poor regime, and lead to inaccurate assessment of users' utilities and suboptimal assortment recommendation. Hence, by adding random projection to Benchmark 3 so that features outside of the significant feature set identified by the Lasso can be recycled and reused to improve users' utility assessment, Lasso-RP-MNL performs the best among all algorithms.

Finally, it is worth mentioning that Benchmark 2 seems to perform well at the beginning but fails to converge in the experiment. Specifically, we first observe that Benchmark 2 performs exceptionally well under very limited samples. To explain, note that with limited samples, estimating a large number of parameters will inevitably lead to high variances and poor estimates. Hence, by projecting high-dimensional data into a lower dimension, Benchmark 2 could significantly reduce the number of parameters that need to be estimated and improve the estimation accuracy, which in turn enables better assortment recommendations. Yet, in contrast to other algorithms that will asymptotically learn the true parameter values, Benchmark 2 suffers from information loss in the process of projecting high-dimensional data into a low-dimensional space, which cannot be corrected asymptotically. Hence, as the sample size T increases, the cumulative regret of Benchmark 2 will eventually exceed Benchmark 3, Benchmark 1, and MNL-Bandit sequentially, as shown in Figure 1.

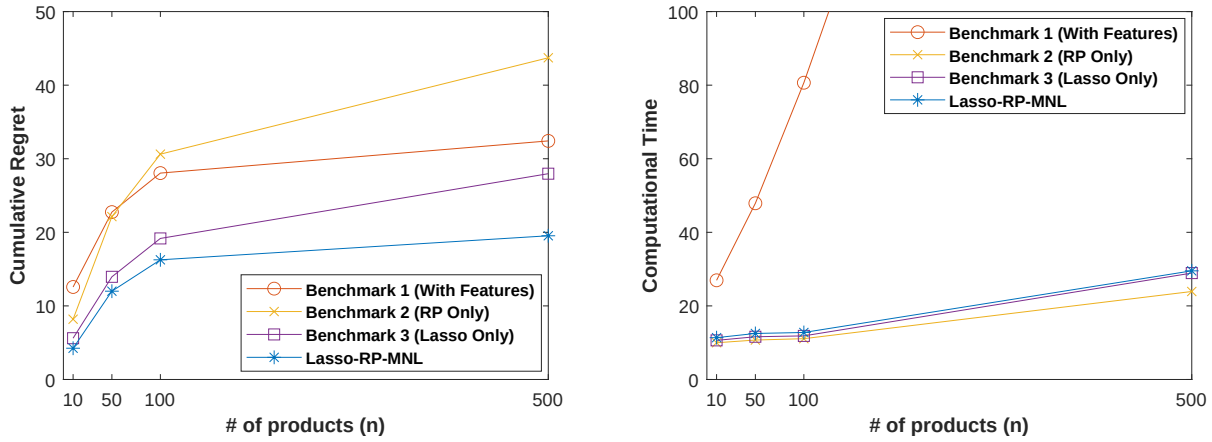
5.2. The Impacts of N, d, and m on Lasso-RP-MNL

In this subsection, we examine the influences of the number of products in the candidate set N, the feature dimension d, and the projection dimension m. Recall that in the first experiment in §5.1, we consider homogeneous users and restrict the product candidate set to be small and remain unchanged for all users so that MNL-Bandit can be included as a benchmark to illustrate the value of contextual information. In this subsection, however, we simulate the real practice environment, where the decision-maker faces heterogeneous users and available products can be innumerable and change from user to user.

To this end, we largely follow the data generation and parameter inputs discussed in §5.1, except for the process of generating the feature vectors for user-product pairs. In particular, for each arriving user, we will regenerate the feature vectors for N user-product pairs (i.e., x_i for $i = \{1, 2, ..., N\}$) by independently and identically drawing them from a Gaussian distribution. Hence, the changes in user-product pairs can reflect the changes in both users' features (i.e., heterogeneous users) and products' features (i.e., a different product candidate set). With heterogeneous users and changing product candidate sets, we will benchmark the Lasso-RP-MNL algorithm against Benchmark 1, 2, and 3 in the next three synthetic experiments.

5.2.1. Impact of the size of the product candidate set N: To examine the impact of the number of products in the candidate set, we vary $N = \{10, 50, 100, 500\}$ while keeping parameters $s = 5$, $d = 50$, $K = 5$, and $m = 5$ unchanged. For different values of N, our proposed Lasso-RP-MNL algorithm always converges before 500 users, so we present the cumulative regret performance and the computational time for Lasso-RP-MNL and Benchmarks 1, 2, and 3 at time $T = 500$ in Figure 2.

Figure 2 **The impact of N on the cumulative regret and the computational time, where $T = 500$ $s = 5$, $d = 50$, $K = 5$, and $m = 5$.**



(a) Cumulative regret

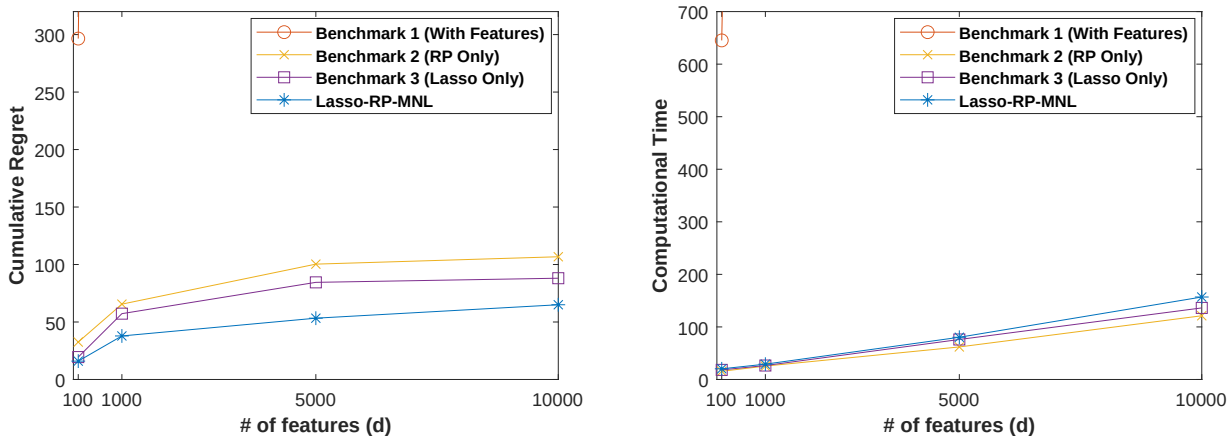
(b) Average computational time per trial

Figure 2(a) suggests that the cumulative regret for all algorithms seems to increase in the number of products in N. This occurs not because there will be more unknown coefficients that need to be estimated, as the feature dimension d (and its corresponding unknown coefficients) remains unchanged in this experiment. Instead, it occurs due to the fact that before the decision-maker is able to fully learn the true values of these coefficients, the estimation errors will have a higher impact on the cumulative regret for a large number of products. These errors will lead to inaccurate estimation regarding users' utilities, which makes it more difficult for the decision-maker to select the optimal assortment, especially under a large candidate pool. Therefore, we observe that all algorithms' cumulative regret increases in N, but among all algorithms, Lasso-RP-MNL has the the lowest cumulative regret, which also tends to grow at the slowest pace.

We also observe that algorithms with dimension reduction techniques (i.e., Benchmark 2, Benchmark 3, and Lasso-RP-MNL) are much more computationally efficient compared to the algorithm without such techniques (i.e., Benchmark 1). Specifically, Figure 2(b) presents the influence of N on the average computational time per trial in seconds. Using the Lasso and/or random projection as dimension reduction techniques could significantly scales down the computational time for estimating parameters and calculating users' utilities for all products, and therefore Benchmark 2, Benchmark 3, and Lasso-RP-MNL perform much better than Benchmark 1, regardless of the size of the candidate set.

5.2.2. Impact of the feature dimension d: Next, we examine the influence of the feature dimension by varying $d = \{100, 1000, 5000, 10000\}$ while keeping parameters $s = 5$, $N = 100$, $K = 5$, and $m = 5$ unchanged. Similarly to the second synthetic experiment, we report the cumulative regret and computational time for Lasso-RP-MNL, Benchmark 1, 2, and 3 at $T = 500$ in Figure 3.

Figure 3 **The impact of d on the cumulative regret and the computational time, where $T = 500$ $s = 5$, $N = 100$, $K = 5$, and $m = 5$.**



(a) Cumulative Regret (b) Average Computational Time per Trial

We first observe that compared to other algorithms, Benchmark 1's cumulative regret and computational time grow dramatically and rapidly out of the chart, as the feature dimension d exceeds 100 in the experiment[2]. This is as expected. When we expand the feature dimension without using any dimension reduction techniques, Benchmark 1 will require a larger number of available samples to achieve reasonable estimation accuracy. Yet, as we increase the feature dimension while keeping the sample size unchanged at $T = 500$, the regret performance of Benchmark 1 inevitably suffers. Further, note that Benchmark 1 will need to estimate coefficients for all features. Consequently, as the feature dimension increases, its computational time surges.

Adopting dimension reduction techniques, Lasso-RP-MNL and the remaining two benchmarks have much lower cumulative regret performance, compared to Benchmark 1. Furthermore, we discovered that as the feature dimension d increases, the cumulative regret for these three algorithms grows. Similarly to previous experiments, our proposed Lasso-RP-MNL has the lowest regret performance, and Benchmark 3 performs better than Benchmark 2. From the computational time's perspective, Benchmark 2, Benchmark 3, and Lasso-RP-MNL all seem to be computationally efficient, while Benchmark 2 maintains a slight advantage over Benchmark 3 and Lasso-RP-MNL.

5.2.3. Impact of the Projection Dimension m: In the final synthetic data experiment, we explore the influences of the projection dimension by varying $m = \{1, 2, 3, 4, 5, 10, 20, 50, 100, 200\}$ and keeping parameters $d = 200$, $s = 5$, $N = 30$, and $K = 5$ unchanged. The cumulative regret and computational time for Lasso-RP-MNL at $T = 500$ are presented in Figure 4. As expected, we first observe that the computational time for Lasso-RP-MNL increases in the projection dimension m.

[2] In our experiments, when the feature dimension d equals $1,000$, the computational time for a single trial of Benchmark 1 will easily exceed one hour. We therefore only plot Benchmark 1's results for $d = 100$ in Figure 1.

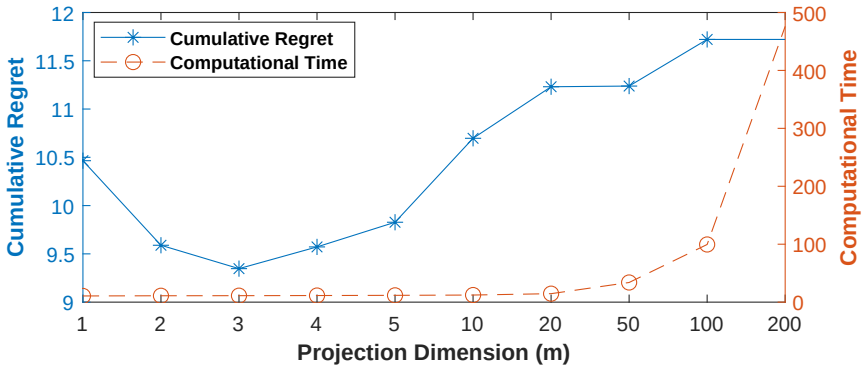


Figure 4 **The impact of m on the cumulative regret and the computational time, where $d = 200$, $s = 5$, $N = 30$, $K = 5$, and $T = 500$.**

The cumulative regret of Lasso-RP-MNL, however, exhibits an unimodal property with respect to the projection dimension m in our experiment. This occurs due to two competing forces (information loss and estimation accuracy), as we increase the projection dimension. Specifically, recall that projecting high-dimensional features into a low dimension will inevitably result in information loss: The lower the projection dimension m is, the greater information loss will be (see Theorem 2, which illustrates the true feature coefficient vector differs significantly from its projected counterpart for a small m). Hence, increasing the projection dimension m will limit information loss. On the other hand, increasing the projection dimension m simultaneously increases the number of unknown coefficients that need to be estimated, which will certainly lead to higher variances and poorer estimates.

When m is very small, a slight increase in the projection dimension m could significantly mitigate information loss. At the same time, because there is a sufficient number of samples at $T = 500$, the negative influence of such a slight increase in the number of unknown parameters does not seem to have a substantial effect on the estimation accuracy. Hence, the benefits of reducing information loss dominant, and increasing the projection dimension decreases the cumulative regret. However, when m is not too small, the above relationships are reversed; and increasing the projection dimension leads to high variances and poor estimates and worsens the cumulative regret performance.

In all experiments, we observe that the projection dimension that minimizes the cumulative regret is typically fairly small compared to the feature dimension. In fact, we can show that if the projection dimension m is chosen to be on the order of $\mathcal{O}(\log d)$, then the Gaussian random projection can be confined to a fixed distortion (see the proof of Lemma 2). In Figure 4, for the feature dimension of 200, we merely need to set the projection dimension to be 3 to minimize Lasso-RP-MNL's cumulative regret performance.

5.3. XianYu Assortment Recommendation Experiment

Finally, we consider a high-dimensional assortment recommendation problem faced by XianYu in practice. To ensure that our experiment is manageable for a single PC, we trimmed the original XianYu dataset, which consists of more than 2 billion features for the user-product pair from 4 million samples (e.g., assortments offered to users and their corresponding responses), to include only 10 thousands features[3] that appear with the highest frequency in these samples.

In practice, for each arriving user, XianYu will first pre-select a personalized product candidate set. Typically, to offer an assortment for an arriving user, there are approximately more than 1 billion available products for XianYu to choose from. Yet, assessing the user's utilities for all available products to identify the optimal assortment is computationally infeasible under the online setting, where the user expects less than half a second delay. Therefore, depending on the arriving user's certain characteristics (such as searching key words, current browsing page, demographics, etc.), XianYu will first pre-select $1,000$ highly correlated products from all available products using its efficient recall mechanisms. Then, XianYu's assortment optimization algorithm – the Top-K algorithm – will pick 20 products from the pre-selected $1,000$ products and offer them to the user.

In this experiment, we include XianYu's Top-K algorithm as another benchmark algorithm. The Top-K algorithm is a hybrid online-offline algorithm: The assessment for each user's choice probability and the assortment optimization are performed online, but the parameter estimation and updating are performed offline. In the Top-K algorithm, XianYu treats each product in an assortment separately and uses a logistic regression to individually assess the user's selection probability with respect to each product. Specifically, for each arriving user, XianYu will first assess this user's selection probabilities, based on the user-product feature pair via logistic regression, for all products in the pre-selected candidate set of $1,000$ products, then calculate the user's expected reward for these products separately, and finally offer the top $K = 20$ products with the highest expected reward as the assortment to this user. In practice, XianYu periodically (typically every couple of hours) updates its estimates for unknown coefficients in the logistic regression via the maximum likelihood estimation. In our experiment, we allow XianYu to update its coefficients at a more frequent rate (i.e., at the same frequency as the Lasso updates in the Lasso-RP-MNL algorithm).

At XianYu, the $1,000$ pre-selected products vary significantly from user to user. Therefore, to simulate such a dynamic environment in the experiment, for each arriving user, we randomly select $1,000$ products from the candidate set of $20,000$ high frequency products in the dataset and then use different assortment optimization algorithms to select 20 products for the user. Finally, we use the actual asking prices for these products as the reward that XianYu will receive when a user

[3] We could extend the experiment to include more features, but doing so would not qualitatively change our results and insights but would considerably increase the computational burden.

clicks/buys an recommended product, and the underlying user's choice model is estimated using the original untrimmed dataset.

In this experiment, we assess the lost revenue under Benchmark 2, Benchmark 3, Lasso-RP-MNL, and the Top-K algorithm by comparing them to an oracle policy. Note that we are unable to include Benchmark 1 in this experiment, because a single trial of Benchmark 1 would take more than 24 hours to finish. It is worth mentioning that as the true coefficient vector is unknown, the "true" oracle policy is impossible to implement in our experiment. Therefore, the oracle policy represents the scenario in which XianYu already has access to all sample data in the original untrimmed dataset to estimate the unknown coefficient vector and identify the optimal assortment accordingly. For each algorithm, we perform 60 trials and report the average loss of revenue for the first $5,000$ users. The computational time for each algorithm per trial is as follows: 259 seconds for the Top-K algorithm[4], 349 seconds for Benchmark 2, 362 seconds for Benchmark 3, and 678 seconds for Lasso-RP-MNL.

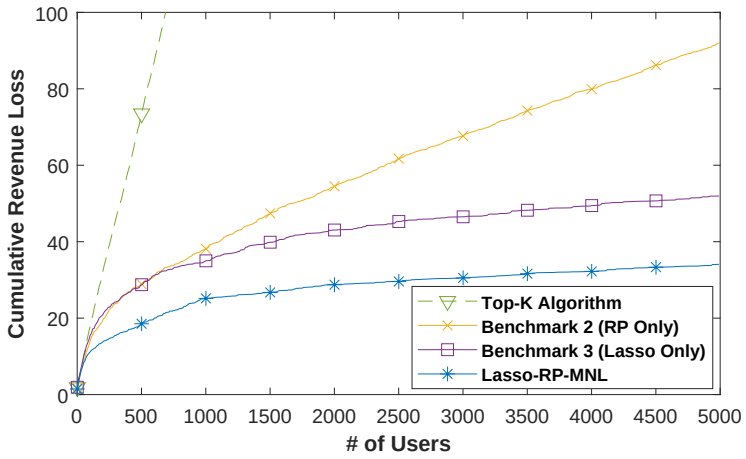


Figure 5 **XianYu assortment recommendation experiment where** $T = 5,000$**,** $d = 10,000$ $N = 1,000$**,** $K = 20$**, and** $m = 10$**.**

Figure 5 plots the cumulative revenue loss (compared to the oracle policy) under Benchmark 2, Benchmark 3, Lasso-RP-MNL, and the Top-K algorithm. In this experiment, the Top-K algorithm and Benchmark 2 seem to have an advantage in terms of computational time (i.e., 259 seconds and 349 seconds per trial, respectively), but both algorithms fail to converge with $5,000$ users and lead to significant revenue loss. In contrast, Benchmark 3 and Lasso-RP-MNL algorithms require more computational efforts, but they have much lower revenue loss and are able to converge with less

[4] In practice, the Top-K algorithm updates its coefficient estimation in an offline fashion, so the computational time reported for the Top-K algorithm excludes the coefficient estimation time to reflect such a practice.

than 2,000 users. Among all algorithms, the Lasso-RP-MNL algorithm performs the best in terms of cumulative revenue loss.

Finally, we believe that the additional computational burden in Lasso-RP-MNL, compared to Top-K, is not a serious concern and can be alleviated by using parallel computing techniques and approximation algorithms. For example, for each arriving user, the decision-maker can compute attraction parameters' bounds for all products in parallel via the parameter-server architecture (Li et al. 2014), which holds the parameter $\hat{\theta}$ and distributes it to all workers that compute bounds simultaneously. Within each worker, we may further deploy the accelerated gradient descent method as the approximation algorithm to reduce the computational time. In addition, as mentioned in §4.4, the Lasso updates and the assortment optimizations can be computed in parallel to further trim down the computational time; or, similar to the Top-K algorithm, the Lasso updates can be executed in a offline fashion. At last, instead of using Gaussian random projection matrix, the decision-maker can adopt the sparse random projection matrix discussed in §4.2 to further improve the computational time.

6. Conclusion

In this paper, we propose a computationally efficient Lasso-RP-MNL algorithm for online assortment optimization problems in high-dimensional settings. This algorithm periodically uses the Lasso to identify significant features that have strong influences on users' choices and adopts random projection to reduce the high-dimensional contextual information to a low-dimensional space. Therefore, the learning and parameter estimation can be performed in low-dimensional fashion to significantly trim down the computational time while maintaining high accuracy in predicting users' utilities and choices. For each arriving user, the Lasso-RP-MNL constructs an upper-confidence bound for every product's attraction parameter, based on which the optimistic assortment can be identified through solving a reformulated linear programming problem.

We demonstrate that the Lasso-RP-MNL algorithm is asymptotically upper-bounded by $\tilde{\mathcal{O}}(\sqrt{T}\log d)$, which achieves logarithmic dependence on the feature dimension d. This improvement is particularly significant for regret performance under high-dimensional settings, where the feature dimension is much larger than the sample size dimension. Furthermore, we show that the Lasso-RP-MNL algorithm continues to perform well even under the data-poor regime, where available samples are extremely limited, with an upper bound of $\tilde{\mathcal{O}}(T^{\frac{2}{3}}\log d)$. Finally, through synthetic-data-based experiments and a high-dimensional XianYu assortment recommendation experiment, we show that compared to existing state-of-the-art algorithms in the literature and industrial practices, the Lasso-RP-MNL algorithm is computationally efficient and can significantly improve the decision-maker's regret performance.

References

Agrawal S, Avadhanula V, Goyal V, Zeevi A (2017) Thompson sampling for the mnl-bandit. *arXiv preprint arXiv:1706.00977* .

Agrawal S, Avadhanula V, Goyal V, Zeevi A (2019) Mnl-bandit: A dynamic learning approach to assortment selection. *Operations Research* 67(5):1453–1485.

Anderson SP, De Palma A, Thisse JF (1992) *Discrete choice theory of product differentiation* (MIT press).

Auer P (2002) Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research* 3(Nov):397–422.

Bastani H, Bayati M (2015) Online decision-making with high-dimensional covariates. *Available at SSRN 2661896* .

Belloni A, Chernozhukov V, et al. (2013) Least squares after model selection in high-dimensional sparse models. *Bernoulli* 19(2):521–547.

Bernstein F, Modaresi S, Sauré D (2018) A dynamic clustering approach to data-driven assortment personalization. *Management Science* 65(5):2095–2115.

Bickel PJ, Ritov Y, Tsybakov AB, et al. (2009) Simultaneous analysis of lasso and dantzig selector. *The Annals of Statistics* 37(4):1705–1732.

Blanchet J, Gallego G, Goyal V (2016) A markov chain approximation to choice modeling. *Operations Research* 64(4):886–905.

Bühlmann P, Van De Geer S (2011) *Statistics for high-dimensional data: methods, theory and applications* (Springer Science & Business Media).

Candes E, Tao T, et al. (2007) The dantzig selector: Statistical estimation when p is much larger than n. *The annals of Statistics* 35(6):2313–2351.

Chen X, Wang Y, Zhou Y (2018) Dynamic assortment optimization with changing contextual information. *arXiv preprint arXiv:1810.13069* .

Cheung WC, Simchi-Levi D (2017) Thompson sampling for online personalized assortment optimization problems with multinomial logit choice models. *Available at SSRN 3075658* .

Clarkson KL, Woodruff DP (2017) Low-rank approximation and regression in input sparsity time. *Journal of the ACM (JACM)* 63(6):54.

Dani V, Hayes TP, Kakade SM (2008) Stochastic linear optimization under bandit feedback .

Davis J, Gallego G, Topaloglu H (2013) Assortment planning under the multinomial logit model with totally unimodular constraint structures. *Work in Progress* .

Fan J, Li R (2001) Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association* 96(456):1348–1360.

Farias VF, Jagabathula S, Shah D (2013) A nonparametric approach to modeling choice with limited data. *Management science* 59(2):305–322.

Feldman JB, Topaloglu H (2017) Revenue management under the markov chain choice model. *Operations Research* 65(5):1322–1342.

Fern XZ, Brodley CE (2003) Random projection for high dimensional data clustering: A cluster ensemble approach. *Proceedings of the 20th international conference on machine learning (ICML-03)*, 186–193.

Filippi S, Cappe O, Garivier A, Szepesvári C (2010) Parametric bandits: The generalized linear case. *Advances in Neural Information Processing Systems*, 586–594.

Gallego G, Wang R (2014) Multiproduct price optimization and competition under the nested logit model with product-differentiated price sensitivities. *Operations Research* 62(2):450–461.

Ghashami M, Liberty E, Phillips JM, Woodruff DP (2016) Frequent directions: Simple and deterministic matrix sketching. *SIAM Journal on Computing* 45(5):1762–1792.

Johnson WB, Lindenstrauss J (1984) Extensions of lipschitz mappings into a hilbert space. *Contemporary mathematics* 26(189-206):1.

Kallus N, Udell M (2016) Dynamic assortment personalization in high dimensions. *arXiv preprint arXiv:1610.05604* .

Kök AG, Fisher ML (2007) Demand estimation and assortment optimization under substitution: Methodology and application. *Operations Research* 55(6):1001–1021.

Kök AG, Fisher ML, Vaidyanathan R (2015) Assortment planning: Review of literature and industry practice. *Retail supply chain management*, 175–236 (Springer).

Kuzborskij I, Cella L, Cesa-Bianchi N (2018) Efficient linear bandits through matrix sketching. *arXiv preprint arXiv:1809.11033* .

Lee JD, Sun DL, Sun Y, Taylor JE, et al. (2016) Exact post-selection inference, with application to the lasso. *The Annals of Statistics* 44(3):907–927.

Li G, Rusmevichientong P, Topaloglu H (2015) The d-level nested logit model: Assortment and price optimization problems. *Operations Research* 63(2):325–342.

Li L, Lu Y, Zhou D (2017) Provably optimal algorithms for generalized linear contextual bandits. *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 2071–2080 (JMLR. org).

Li M, Andersen DG, Park JW, Smola AJ, Ahmed A, Josifovski V, Long J, Shekita EJ, Su BY (2014) Scaling distributed machine learning with the parameter server. *11th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 14)*, 583–598.

Li P, Hastie TJ, Church KW (2006) Very sparse random projections. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 287–296 (ACM).

Loh PL, Wainwright MJ (2013) Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. *Advances in Neural Information Processing Systems*, 476–484.

Luo H, Agarwal A, Cesa-Bianchi N, Langford J (2016) Efficient second order online learning by sketching. *Advances in Neural Information Processing Systems*, 902–910.

Mahajan S, Van Ryzin G (2001) Stocking retail assortments under dynamic consumer substitution. *Operations Research* 49(3):334–351.

Mahajan S, van Ryzin GJ (1999) Retail inventories and consumer choice. *Quantitative models for supply chain management*, 491–551 (Springer).

Matoušek J (2008) On variants of the johnson–lindenstrauss lemma. *Random Structures & Algorithms* 33(2):142–156.

Matoušek J (2013) Lecture notes on metric embeddings. Technical report, Technical report, ETH Zürich.

McFadden D (1980) Econometric models for probabilistic choice among products. *Journal of Business* S13–S29.

McFadden D, et al. (1973) Conditional logit analysis of qualitative choice behavior .

Meinshausen N, Bühlmann P, et al. (2006) High-dimensional graphs and variable selection with the lasso. *The annals of statistics* 34(3):1436–1462.

Meinshausen N, Yu B, et al. (2009) Lasso-type recovery of sparse representations for high-dimensional data. *The annals of statistics* 37(1):246–270.

Netessine S, Rudi N (2003) Centralized and competitive inventory models with demand substitution. *Operations research* 51(2):329–335.

Oh Mh, Iyengar G (2019a) Multinomial logit contextual bandits .

Oh Mh, Iyengar G (2019b) Thompson sampling for multinomial logit contextual bandits. *Advances in Neural Information Processing Systems*, 3145–3155.

Pilanci M, Wainwright MJ (2015) Randomized sketches of convex programs with sharp guarantees. *IEEE Transactions on Information Theory* 61(9):5096–5115.

Rusmevichientong P, Shen ZJM, Shmoys DB (2010) Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Operations research* 58(6):1666–1680.

Rusmevichientong P, Van Roy B, Glynn PW (2006) A nonparametric approach to multiproduct pricing. *Operations Research* 54(1):82–98.

Ryzin Gv, Mahajan S (1999) On the relationship between inventory costs and variety benefits in retail assortments. *Management Science* 45(11):1496–1509.

Sauré D, Zeevi A (2013) Optimal dynamic assortment planning with demand learning. *Manufacturing & Service Operations Management* 15(3):387–404.

Smith SA, Agrawal N (2000) Management of multi-item retail inventory systems with demand substitution. *Operations Research* 48(1):50–64.

Tibshirani R (1996) Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* 58(1):267–288.

Tropp JA, et al. (2015) An introduction to matrix concentration inequalities. *Foundations and Trends® in Machine Learning* 8(1-2):1–230.

Vershynin R (2010) Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027* .

Wang X, Wei MM, Yao T (2018) Online learning and decision-making under generalized linear model with high-dimensional data. *Available at SSRN 3294832* .

Zhang CH, et al. (2010) Nearly unbiased variable selection under minimax concave penalty. *The Annals of statistics* 38(2):894–942.

Electronic Companion to "Online Assortment Optimization with High-Dimensional Data"

EC.1. Proof of Lemma 1

Since $\hat{\beta}$ is the optimal solution for the Lasso problem, we have

$$\nabla L(\hat{\beta}) + \lambda \partial \|\hat{\beta}\|_1 = 0,$$

where $\partial(\cdot)$ denotes the subgradient. As the negative log-likelihood function L is twice differentiable, there exists a ξ such that

$$\begin{aligned}
&\nabla L(\beta^*) - \nabla^2 L(\xi)(\beta^* - \hat{\beta}) = \nabla L(\hat{\beta}) \\
\Rightarrow &\nabla L(\beta^*) - \nabla^2 L(\xi)(\beta^* - \hat{\beta}) + \lambda \partial \|\hat{\beta}\|_1 = 0 \\
&\Rightarrow (\beta^* - \hat{\beta})^T \nabla^2 L(\xi)(\beta^* - \hat{\beta}) = (\beta^* - \hat{\beta})^T \left(\nabla L(\beta^*) + \lambda \partial \|\hat{\beta}\|_1 \right) \\
&\Rightarrow (\beta^* - \hat{\beta})^T \nabla^2 L(\xi)(\beta^* - \hat{\beta}) \leq \|\beta^* - \hat{\beta}\|_1 \left(\|\nabla L(\beta^*)\|_\infty + \lambda \right). \qquad \text{(EC.1)}
\end{aligned}$$

Next, we will build the lower bound for the left-hand-side of (EC.1) by the following technical lemma. (The proofs for all technical lemmas are deferred to the end of this Electronic Companion.)

Lemma EC.1. *Denote n as the random sample size up to time T. If Assumption A.2 holds, then the following inequality holds With probability $1 - \exp(-Cn)$:*

$$u^T \nabla^2(\xi | x, \mathcal{A}) u \geq \frac{K(K-1)\kappa}{4s} \|u_S\|_1^2, \qquad \text{(EC.2)}$$

where u satisfy $\|u_{S^c}\|_1 \leq 3\|u_S\|_1$ and $C = \frac{1}{2}K(K-1)\left(\kappa/256 s x_{\max}^2 (3 + 2\sqrt{2}(1 + 2x_{\max})\right)^2$. Moreover, when $n > \log T / C$, we have

$$\mathbb{P}\left(u^T \nabla^2(\xi | x, \mathcal{A}) u \geq \frac{K(K-1)\kappa}{4s} \|u_S\|_1^2 \right) \geq 1 - \frac{1}{T} \qquad \text{(EC.3)}$$

Note that based on Lemma EC.1, to show that the left-hand-side of (EC.1) is lower bounded by $K(K-1)\kappa/(4s) \cdot \|\beta^*_{\mathcal{S}^*} - \hat{\beta}_{\mathcal{S}^*}\|_1^2$, we only need to prove $\|\beta^*_{(\mathcal{S}^*)^c} - \hat{\beta}_{(\mathcal{S}^*)^c}\|_1 \leq 3\|\beta^*_{\mathcal{S}^*} - \hat{\beta}_{\mathcal{S}^*}\|_1$. As $\hat{\beta}$ is the optimal solution for the Lasso problem, we have

$$\begin{aligned}
&L(\hat{\beta}) + \lambda \|\hat{\beta}\|_1 \leq L(\beta^*) + \lambda \|\beta^*\|_1 \\
&\Rightarrow L(\hat{\beta}) - L(\beta^*) \leq \lambda(\|\beta^*\|_1 - \|\hat{\beta}\|_1) \\
&\Rightarrow \nabla L(\beta^*)(\hat{\beta} - \beta^*) \leq \lambda \left(\|\beta^*\|_1 - \|\hat{\beta}\|_1 \right) \\
&\Rightarrow -\|\nabla L(\beta^*)\|_\infty \|\hat{\beta} - \beta^*\|_1 \leq \lambda \left(\|\beta^*\|_1 - \|\hat{\beta}\|_1 \right) \\
\Rightarrow &-\|\nabla L(\beta^*)\|_\infty (\|\hat{\beta}_{(\mathcal{S}^*)^c} - \beta^*_{(\mathcal{S}^*)^c}\|_1 + \|\hat{\beta}_{\mathcal{S}^*} - \beta^*_{\mathcal{S}^*}\|_1) \leq \lambda(\|\hat{\beta}^*_{\mathcal{S}^*}\|_1 + 0 - \|\hat{\beta}_{\mathcal{S}^*}\|_1 - \|\hat{\beta}_{(\mathcal{S}^*)^c}\|_1) \\
\Rightarrow &-\|\nabla L(\beta^*)\|_\infty (\|\hat{\beta}_{(\mathcal{S}^*)^c} - \beta^*_{(\mathcal{S}^*)^c}\|_1 + \|\hat{\beta}_{\mathcal{S}^*} - \beta^*_{\mathcal{S}^*}\|_1) \leq \lambda(\|\hat{\beta}^*_{\mathcal{S}^*} - \hat{\beta}_{\mathcal{S}^*}\|_1 - \|\hat{\beta}^*_{(\mathcal{S}^*)^c} - \hat{\beta}_{(\mathcal{S}^*)^c}\|_1) \\
&\Rightarrow (\lambda - \|\nabla L(\beta^*)\|_\infty) \|\hat{\beta}_{(\mathcal{S}^*)^c} - \beta^*_{(\mathcal{S}^*)^c}\|_1 \leq (\lambda + \|\nabla L(\beta^*)\|_\infty) \|\hat{\beta}^*_{\mathcal{S}^*} - \hat{\beta}_{\mathcal{S}^*}\|_1. \qquad \text{(EC.4)}
\end{aligned}$$

Therefore, if we have $\|\nabla L(\beta^*)\|_\infty \le \frac{1}{2}\lambda$, then (EC.4) directly implies $\|\beta^*_{(\mathcal{S}^*)^c} - \hat{\beta}_{(\mathcal{S}^*)^c}\|_1 \le 3\|\beta^*_{\mathcal{S}^*} - \hat{\beta}_{\mathcal{S}^*}\|_1$. Such a condition can be shown in the following technical lemma.

LEMMA EC.2. *Let n denote the random sample size. If Assumption A.1 holds, then for any $T > 0$, we have*

$$\mathbb{P}\left(\|\nabla L(\beta^*)\|_\infty \ge \sqrt{\frac{2x_{\max}^2(\log d + \log T)}{n}}\right) \le \frac{2}{T}. \tag{EC.5}$$

If we set $\lambda = 2\sqrt{\frac{2x_{\max}^2(\log d+\log T)}{n}}$, then Lemma EC.2 suggests that with probability $1 - O(T^{-1})$, we have

$$\|\nabla L(\beta^*)\|_\infty \le \frac{1}{2}\lambda. \tag{EC.6}$$

Combining (EC.4) and (EC.6), we have

$$\|\hat{\beta}_{(\mathcal{S}^*)^c} - \beta^*_{(\mathcal{S}^*)^c}\|_1 \le 3\|\hat{\beta}^*_{\mathcal{S}^*} - \hat{\beta}_{\mathcal{S}^*}\|_1. \tag{EC.7}$$

Hence, We can use Lemma EC.1 to show that with high probability, the following inequality holds:

$$(\beta^* - \hat{\beta})^T \nabla^2 L(\xi)(\beta^* - \hat{\beta}) \ge \frac{K(K-1)\kappa}{4s}\|\beta^*_{\mathcal{S}^*} - \hat{\beta}_{\mathcal{S}^*}\|_1^2. \tag{EC.8}$$

Using (EC.7) and (EC.8) we can show

$$(\beta^* - \hat{\beta})^T \nabla^2 L(\xi)(\beta^* - \hat{\beta}) \ge \frac{K(K-1)\kappa}{16s}\|\beta^* - \hat{\beta}\|_1^2. \tag{EC.9}$$

Moreover, combine (EC.9), (EC.6) and (EC.1) and we reach

$$\begin{aligned} \frac{K(K-1)\kappa}{16s}\|\beta^* - \hat{\beta}\|_1^2 &\le \frac{3}{2}\lambda\|\beta^* - \hat{\beta}\|_1 \\ \Rightarrow \|\beta^* - \hat{\beta}\|_1 &\le \frac{24s}{K(K-1)\kappa}\lambda \\ \Rightarrow \|\beta^* - \hat{\beta}\|_2 &\le \frac{24s}{K(K-1)\kappa}\lambda, \end{aligned} \tag{EC.10}$$

where we use $\|\cdot\|_2 \le \|\cdot\|_1$ in the last inequality . The remaining part of this Lemma follows directly by setting $C_{lasso} = \frac{48\sqrt{2}x_{\max}}{K(K-1)\kappa}$ and $n \ge \log T/C = \mathcal{O}(s^2 \log T)$.

EC.2. Proof of Theorem 1

When event $\mathcal{E}_{lasso}(T)$ holds, we have

$$\|\hat{\beta} - \beta^*\| \le \mathcal{G}_0(T),$$

which implies that $|\hat{\beta}_j| \le |\beta^*_j| + \mathcal{G}_0(T)$ and $|\hat{\beta}_j| \ge |\beta^*_j| - \mathcal{G}_0(T)$ for any j.

Combining $|\hat{\beta}_j| \le |\beta^*_j| + \mathcal{G}_0(T)$ with the definition of the index set $\mathcal{S} = \{j : |\hat{\beta}_j| \ge 2\mathcal{G}_0(T)\}$, we can show that

$$j \notin \mathcal{S} \Rightarrow |\hat{\beta}_j| < 2\mathcal{G}_0(T) \Rightarrow |\beta^*_j| < 3\mathcal{G}_0(T). \tag{EC.11}$$

If $\mathcal{G}_0(T) < \frac{1}{3}\beta_{\min}$, then (EC.11) implies that for any $j \notin \mathcal{S}$,

$$|\beta^*_j| < \beta_{\min} \Rightarrow \beta^*_j = 0. \tag{EC.12}$$

Similarly, combining $|\hat{\beta}_j| \le |\beta_j^*| + \mathcal{G}_0(T)$ with the definition of the index set $\mathcal{S}$, we have

$$j \in \mathcal{S} \Rightarrow |\hat{\beta}_j| \ge 2\mathcal{G}_0(T) \Rightarrow |\beta_j^*| \ge \mathcal{G}_0(T) > 0 \Rightarrow \mathcal{S} \subseteq \mathcal{S}^* \Rightarrow |\mathcal{S}| \le |\mathcal{S}^*| = s. \tag{EC.13}$$

EC.3. Proof of Lemma 2

See the proof of Lemma 2.3.1 in Matoušek 2013.

EC.4. Proof of Theorem 2

As Q is the permutation matrix, we have $Q^T Q = I$. Hence, we can show that

$$\begin{aligned}
\|(I-\Sigma)\beta^*\| &= \|Q^T(I - P_0^T P_0)Q\beta^*\| \\
&= \sqrt{(\beta^*)^T Q^T (I - P_0^T P_0)^T Q Q^T (I - P_0^T P_0) Q\beta^*} \\
&= \sqrt{(\beta^*)^T Q^T (I - P_0^T P_0)^T (I - P_0^T P_0) Q\beta^*} \\
&= \|(I - P_0^T P_0)Q\beta^*\| \\
&= \left\| \left[\begin{pmatrix} I & \\ & I \end{pmatrix} - \begin{pmatrix} I & \\ & P^T P \end{pmatrix} \right] \begin{pmatrix} \beta_{\mathcal{S}}^* \\ \beta_{\mathcal{S}^c}^* \end{pmatrix} \right\| \qquad \text{(EC.14)} \\
&= \|(I - P^T P)\beta_{\mathcal{S}^c}^*\|,
\end{aligned}$$

where (EC.14) comes from the definition and construction of the permutation matrix Q and the projection matrix P_0. As β^* is s-sparse, there will be at most s non-zero coefficients in $\beta_{\mathcal{S}^c}^*$. Without loss of generality, we assume that the first k elements of $\beta_{\mathcal{S}^c}^*$ might be non-zero, and we can show that

$$\begin{aligned}
\|(I - P^T P)\beta_{\mathcal{S}^c}^*\| &= \left\| \begin{pmatrix} I - P_k^T P_k & -P_k^T P_{k^c} \\ -P_{k^c}^T P_k & I - P_{k^c}^T P_{k^c} \end{pmatrix} \begin{pmatrix} \beta_{\mathcal{S}^c,k}^* \\ 0 \end{pmatrix} \right\| \\
&= \left\| \begin{pmatrix} (I - P_k^T P_k)\beta_{\mathcal{S}^c,k}^* \\ -P_{k^c}^T P_k \beta_{\mathcal{S}^c,k}^* \end{pmatrix} \right\| \\
&\le \|(I - P_k^T P_k)\beta_{\mathcal{S}^c,k}^*\| + \|P_{k^c}^T P_k \beta_{\mathcal{S}^c,k}^*\|,
\end{aligned}$$

where we separate the random projection matrix P into two sub-matrices $P_k \in \mathbb{R}^{m \times k}$ and $P_{k^c} \in \mathbb{R}^{m \times (|\mathcal{S}^c| - k)}$ with $P = [P_k \ P_{k^c}]$. Next, we need to separately bound $\|(I - P_k^T P_k)\beta_{\mathcal{S}^c,k}^*\|$ and $\|P_{k^c}^T P_k \beta_{\mathcal{S}^c,k}^*\|$.

- *The bound for* $\|(I - P_k^T P_k)\beta_{\mathcal{S}^c,k}^*\|$:

The Remark 5.40 in Vershynin 2010 shows that there exits a constant $C_1 > 0$ such that for any $t > 0$, the following inequality holds with probability $1 - \exp(-C_1 t^2)$:

$$\|P_k^T P_k - I\| \le \max\{\delta, \delta^2\},$$

where P_k is a sub-matrix of P with k columns and

$$\delta = C_1 \sqrt{\frac{k}{m}} + \frac{t}{\sqrt{m}}. \tag{EC.15}$$

Combining this result with Theorem 1 and using the union bound, we can show that for any k dimensional subspace of the original d dimensional space, if $k < s$, then the following inequality holds with probability at least $1 - \binom{d}{s}\exp(-C_1 t^2) \le 1 - d^s \exp(-C_1 t^2) = 1 - \exp(-C_1 t^2 + s\log d)$:

$$\|(I - P_k^T P_k)\beta^*_{\mathcal{S}^c,k}\| \le 3\max\{\delta, \delta^2\}\sqrt{s}\mathcal{G}_0(T). \tag{EC.16}$$

Further, if we set $t = \sqrt{\log T/C_1 + s\log d}$, then we can immediately show that

$$1 - \exp(-C_1 t^2 + s\log d) \ge 1 - \frac{1}{T}, \tag{EC.17}$$

and

$$\begin{aligned} &\delta = (C_1\sqrt{k} + \sqrt{\log T/C_1 + s\log d})/\sqrt{m} \\ \Rightarrow &\delta^2 \le 2(C_1^2 k + \log T/C_1 + s\log d)/m, \end{aligned} \tag{EC.18}$$

where (EC.18) uses the fact that $(\sqrt{a} + \sqrt{b})^2 \le 2a + 2b$ for any $a, b \ge 0$. Finally, when $m \le \log T/C_1 + s\log d$, we have $\delta > 1$, which implies $\delta^2 \ge \delta$. Then via (EC.16), (EC.17) and (EC.18), we can bound $\|(I - P_k^T P_k)\beta^*_{\mathcal{S}^c,k}\|$ with probability $1 - \mathcal{O}(T^{-1})$:

$$\begin{aligned} \|(I - P_k^T P_k)\beta^*_{\mathcal{S}^c,k}\| &\le 3\sqrt{s}\delta^2\mathcal{G}_0(T) \\ &\le 6\sqrt{s}\frac{C_1^2 k + s\log d + \log T/C_1}{m}\mathcal{G}_0(T) \\ &\le 6\sqrt{s}\frac{(C_1^2 + \log d)s + \log T/C_1}{m}\mathcal{G}_0(T), \end{aligned} \tag{EC.19}$$

where we use $k \le s$ in the last inequality.

- *The Bound for* $\|P_{k^c}^T P_k \beta^*_{\mathcal{S}^c,k}\|$:

As matrix P_{k^c} and P_k are filled with i.i.d $N(0, 1/m)$ random Gaussian elements, we apply Lemma 2 twice to have the following two inequalities:

$$\mathbb{P}\left(\|P_{k^c}^T P_k \beta^*_{\mathcal{S}^c,k}\| \ge (1+\epsilon)\|P_k \beta^*_{\mathcal{S}^c,k}\|\right) \le 2\exp(-C_2\epsilon^2 m),$$

and

$$\mathbb{P}\left(\|P_k \beta^*_{\mathcal{S}^c,k}\| \ge (1+\epsilon)\|\beta^*_{\mathcal{S}^c,k}\|\right) \le 2\exp(-C_2\epsilon^2 m).$$

Combining these two inequalities, we can show that

$$\begin{aligned} &\mathbb{P}\left(\|P_{k^c}^T P_k \beta^*_{\mathcal{S}^c,k}\| \ge (1+\epsilon)(1+\epsilon)\|\beta^*_{\mathcal{S}^c,k}\|\right) \le 4\exp(-C_2\epsilon^2 m) \\ \Rightarrow &\mathbb{P}\left(\|P_{k^c}^T P_k \beta^*_{\mathcal{S}^c,k}\| \ge 2(1+\epsilon^2)\|\beta^*_{\mathcal{S}^c,k}\|\right) \le 4\exp(-C_2\epsilon^2 m), \end{aligned} \tag{EC.20}$$

where we use the observation that $(a+b)^2 \le 2(a^2+b^2)$ for all $a, b \in \mathbb{R}$. Then (EC.20) directly implies the following inequality by setting $\epsilon = \sqrt{\frac{\log T}{C_2 m}}$:

$$\mathbb{P}\left(\|P_{k^c}^T P_k \beta^*_{\mathcal{S}^c,k}\| \ge \frac{2m + 2\log T/C_2}{m}\|\beta^*_{\mathcal{S}^c,k}\|\right) \le 4\exp(-\log T) = \mathcal{O}(T^{-1}).$$

Now, combining with the fact that $\|\beta^*_{\mathcal{S}^c,k}\| \le \sqrt{s}\cdot 3\mathcal{G}_0(T)$ from Theorem 1, we can establish the following probability bound for $\|P^T_{k^c}P_k\beta^*_{\mathcal{S}^c,k}\|$:

$$\mathbb{P}\left(\|P^T_{k^c}P_k\beta^*_{\mathcal{S}^c,k}\| \le \frac{6m+6\log T/C_2}{m}\sqrt{s}\mathcal{G}_0(T)\right) \ge 1-\mathcal{O}(T^{-1}). \tag{EC.21}$$

Finally, combining the two bounds developed in (EC.19) and (EC.21), we can show that the following inequality holds with probability $1-\mathcal{O}(T^{-1})$, when $m \le \log T/C_1 + s\log d$:

$$\|(I-\Sigma)\beta^*\| \le \frac{6(C_1^2+\log d)s+\log T(6/C_1+6/C_2)+6m}{m}\sqrt{s}\mathcal{G}_0(T). \tag{EC.22}$$

Moreover, we state in Theorem 1 that when $\mathcal{G}_0(T) < \frac{1}{3}\beta_{\min}$, we have $\beta^*_{\mathcal{S}}=0$, which suggests the no information loss result, i.e.,

$$\|(I-\Sigma)\beta^*\| = 0.$$

EC.5. Proof of Lemma 3

To simplify the notations in this proof, we will ignore the $\mathcal{A}$ subscript in probability term $p_{\cdot,\mathcal{A}}(\cdot)$ and re-define $\mathcal{A}$ as the assortment including both the original assortment $\mathcal{A}$ and the no-purchase option (i.e., $\mathcal{A} := \mathcal{A}\cup\{0\}$), as long as doing so does not cause any misinterpretation. Accordingly, the decision-maker's expected reward for a coefficient vector θ under this re-defined assortment $\mathcal{A}$ can be simplified into $R_{\mathcal{A}}(\theta) = \frac{\sum_{i\in\mathcal{A}} r_i \exp((P_0Qx_i)^T\theta)}{\sum_{i\in\mathcal{A}}\exp((P_0Qx_i)^T\theta)}$.

Using Taylor expansion, we can show that there exists a ξ such that

$$\begin{aligned}
|R_{\mathcal{A}}(\hat\theta) - R_{\mathcal{A}}(P_0Q\beta^*)| &= \left\|\nabla R_{\mathcal{A}}(\hat\theta)^T(\hat\theta - P_0Q\beta^*) - \frac{1}{2}(\hat\theta - P_0Q\beta^*)^T\nabla^2 R_{\mathcal{A}}(\xi)(\hat\theta - P_0Q\beta^*)\right\| \\
&\le \|\nabla R_{\mathcal{A}}(\hat\theta)^T(\hat\theta - P_0Q\beta^*)\| + \left\|\frac{1}{2}(\hat\theta - P_0Q\beta^*)^T\nabla^2 R_{\mathcal{A}}(\xi)(\hat\theta - P_0Q\beta^*)\right\| \\
&\le \sqrt{\|\nabla R_{\mathcal{A}}(\hat\theta)^T(\hat\theta - P_0Q\beta^*)\|^2} + \frac{1}{2}\left\|\nabla^2\mathcal{R}_{\mathcal{A}}(\xi)\right\|_{op}\|\hat\theta - P_0Q\beta^*\|^2 \\
&= \overbrace{\sqrt{(\hat\theta - P_0Q\beta^*)^T\nabla R_{\mathcal{A}}(\hat\theta)\nabla R_{\mathcal{A}}(\hat\theta)^T(\hat\theta - P_0Q\beta^*)}}^{\text{ⓐ}} + \overbrace{\frac{1}{2}\left\|\nabla^2\mathcal{R}_{\mathcal{A}}(\xi)\right\|_{op}\|\hat\theta - P_0Q\beta^*\|^2}^{\text{ⓑ}}.
\end{aligned}$$

Now, we will separately build upper bounds for ⓐ and ⓑ.

Analysis for ⓐ: From the definition of $R_{\mathcal{A}}(\hat\theta)$ and using $p_{Q^TP_0^T\theta}(i)$ to represent the probability of selecting product i from the assortment $\mathcal{A}$ (i.e., $p_{Q^TP_0^T\theta}(i) = 1/(\sum_{i\in\mathcal{A}}\exp((P_0Qx_i)^T\theta))$), we can show that

$$\begin{aligned}
\nabla R_{\mathcal{A}}(\hat\theta) &= \frac{\sum_{i\in\mathcal{A}} r_i\exp(x_i^TQ^TP_0^T\hat\theta)P_0Qx_i\cdot(\sum_{j\in\mathcal{A}}\exp(x_j^TQ^TP_0^T\hat\theta))}{(\sum_{j\in\mathcal{A}}\exp(x_j^TQ^TP_0^T\hat\theta))^2} \\
&\quad - \frac{(\sum_{i\in\mathcal{A}} r_i\exp(x_i^TQ^TP_0^T\hat\theta))\cdot(\sum_{j\in\mathcal{A}}\exp(x_j^TQ^TP_0^T\hat\theta)P_0Qx_j)}{(\sum_{j\in\mathcal{A}}\exp(x_j^TQ^TP_0^T\hat\theta))^2} \\
&= \sum_{i\in\mathcal{A}} P_0Qx_ir_ip_{Q^TP_0^T\hat\theta}(i) - \sum_{i\in\mathcal{A}} r_ip_{Q^TP_0^T\hat\theta}(i)\cdot\sum_{j\in\mathcal{A}} P_0Qx_jp_{Q^TP_0^T\hat\theta}(j) \\
&= \sum_{i\in\mathcal{A}} r_ip_{Q^TP_0^T\hat\theta}(i)\left[P_0Qx_i - \sum_{j\in\mathcal{A}} P_0Qx_jp_{Q^TP_0^T\hat\theta}(j)\right]
\end{aligned}$$

$$= \sum_{i\in\mathcal{A}} p_{Q^T P_0^T \hat{\theta}}(i) \left[r_i - \sum_{j\in\mathcal{A}} p_{Q^T P_0^T \hat{\theta}}(j) r_i \right] \left[P_0 Q x_i - \sum_{j\in\mathcal{A}} p_{Q^T P_0^T \hat{\theta}}(j) P_0 Q x_j \right]. \tag{EC.23}$$

We write (EC.23) in short-hand notation as follows:

$$(\text{EC.23}) := \tilde{\mathbb{E}} \left[\left(r - \tilde{\mathbb{E}}[r] \right) \left(z - \tilde{\mathbb{E}}[z] \right) \right], \tag{EC.24}$$

where $\tilde{\mathbb{E}}$ represents the expectation w.r.t. probability distribution $\{p_{Q^T P_0^T \hat{\theta}}(i)\}$ and $z_i = P_0 Q x_i$ for all $i \in \mathcal{A}$. Then we have

$$\begin{aligned}
\nabla R_S(\hat{\theta}) \nabla R_S(\hat{\theta})^T &= \tilde{\mathbb{E}} \left[\left(r - \tilde{\mathbb{E}}[r] \right) \left(z - \tilde{\mathbb{E}}[z] \right) \right] \cdot \tilde{\mathbb{E}} \left[\left(r - \tilde{\mathbb{E}}[r] \right) \left(z - \tilde{\mathbb{E}}[z] \right) \right]^T \\
&= \tilde{\mathbb{E}} \left[\left(r - \tilde{\mathbb{E}}[r] \right) \left(z - \tilde{\mathbb{E}}[z] \right) \cdot \overbrace{\tilde{\mathbb{E}} \left[\left(r - \tilde{\mathbb{E}}[r] \right) \left(z - \tilde{\mathbb{E}}[z] \right) \right]^T}^{*} \right] && \text{(EC.25)} \\
&\preceq \tilde{\mathbb{E}} \left[\tilde{\mathbb{E}} \left[\left(r - \tilde{\mathbb{E}}[r] \right) \left(z - \tilde{\mathbb{E}}[z] \right) \cdot \left(r - \tilde{\mathbb{E}}[r] \right) \left(z - \tilde{\mathbb{E}}[z] \right)^T \right] \right] && \text{(EC.26)} \\
&= \tilde{\mathbb{E}} \left[\left(r - \tilde{\mathbb{E}}[r] \right)^2 \left(z - \tilde{\mathbb{E}}[z] \right) \left(z - \tilde{\mathbb{E}}[z] \right)^T \right] \\
&\preceq R_{\max}^2 \tilde{\mathbb{E}} \left[\left(z - \tilde{\mathbb{E}}[z] \right) \left(z - \tilde{\mathbb{E}}[z] \right)^T \right],
\end{aligned}$$

where in (EC.26) we uses the Jensen's inequality (e.g., equation 2.2.2 in Tropp et al. 2015) on the * term in (EC.25).

To simplify notation, we can show that $\nabla^2 f_{\mathcal{A}}(\hat{\theta}) = \tilde{\mathbb{E}} \left[\left(z - \tilde{\mathbb{E}}[z] \right) \left(z - \tilde{\mathbb{E}}[z] \right)^T \right]$ (see Lemma EC.8). Then, we have

$$\sqrt{(\hat{\theta} - P_0 Q \beta^*)^T \nabla R_{\mathcal{A}}(\hat{\theta}) \nabla R_{\mathcal{A}}(\hat{\theta})^T (\hat{\theta} - P_0 Q \beta^*)} \le R_{\max} \sqrt{(\hat{\theta} - P_0 Q \beta^*)^T \nabla^2 f_{\mathcal{A}}(\hat{\theta}) (\hat{\theta} - P_0 Q \beta^*)}. \tag{EC.27}$$

Analysis for ⓑ: As we assume $\|\nabla^2 \mathcal{R}_{\mathcal{A}}(\xi)\|_{op} \le \lambda_{\max}$, we can bound ⓑ as follows:

$$\frac{1}{2} \left\| \nabla^2 \mathcal{R}_{\mathcal{A}}(\xi) \right\|_{op} \|\hat{\theta} - P_0 Q \beta^*\|^2 \le \frac{1}{2} \lambda_{\max} \delta^2.$$

Combining the upper bounds for the part ⓐ and part ⓑ, we have

$$\begin{aligned}
&|R_{\mathcal{A}}(\hat{\theta}) - R_{\mathcal{A}}(P_0 Q \beta^*)| \le R_{\max} \sqrt{(\hat{\theta} - P_0 Q \beta^*)^T \nabla^2 f_{\mathcal{A}}(\hat{\theta}) (\hat{\theta} - P_0 Q \beta^*)} + \frac{1}{2} \lambda_{\max} \delta^2 \\
\Rightarrow &|R_{\mathcal{A}}(\hat{\theta}) - R_{\mathcal{A}}(P_0 Q \beta^*)| - \frac{1}{2} \lambda_{\max} \delta^2 \le R_{\max} \sqrt{(\hat{\theta} - P_0 Q \beta^*)^T \nabla^2 f_{\mathcal{A}}(\hat{\theta}) (\hat{\theta} - P_0 Q \beta^*)}
\end{aligned} \tag{EC.28}$$

Denote $H^* = \left(\sum_i^T \nabla^2 f_{\mathcal{A}_i}(P_0 Q \beta^*) \right)$. We show that H^* is positive definite with high probability in Lemma EC.6. Therefore, (EC.28) implies

$$\begin{aligned}
&|R_{\mathcal{A}}(\hat{\theta}) - R_{\mathcal{A}}(P_0 Q \beta^*)| - \frac{1}{2} \lambda_{\max} \delta^2 \\
&\le R_{\max} \sqrt{(\hat{\theta} - P_0 Q \beta^*)^T (H^*)^{1/2} (H^*)^{-1/2} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) (H^*)^{-1/2} (H^*)^{1/2} (\hat{\theta} - P_0 Q \beta^*)} \\
&\le R_{\max} \left\| (\hat{\theta} - P_0 Q \beta^*)^T (H^*)^{1/2} \right\| \sqrt{\left\| (H^*)^{-1/2} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) (H^*)^{-1/2} \right\|_{op}}
\end{aligned}$$

$$= R_{\max}\sqrt{(\hat{\theta} - P_0 Q\beta^*)^T H^* (\hat{\theta} - P_0 Q\beta^*)} \cdot \sqrt{\left\| (H^*)^{-1/2} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) (H^*)^{-1/2} \right\|_{op}}. \tag{EC.29}$$

Then, we use the following Lemma:

LEMMA EC.3. *Let $\delta = \|\theta - P_0 Q\beta^*\|$ and C_3 be a positive constant. Under Assumptions A.1 and A.3, events $\mathcal{E}_2(m,T)$ and $\mathcal{E}_{rp}(m,d,1/2)$, if $\delta \le \min\{\frac{3}{4}\frac{n_T \mu}{T L_3}, \frac{\rho}{8K x_{\max}}\}$ and $\mathcal{G}_1(m,T) \le \frac{\rho}{8K x_{\max}}$, then the following inequality holds for $T \ge 2$ with probability $1 - \mathcal{O}(1/T)$:*

$$(\theta - P_0 Q\beta^*)^T \left(\sum_t \nabla^2 f_{\mathcal{A}_t}(P_0 Q\beta^*) \right) (\theta - P_0 Q\beta^*) \le 16 x_{\max}^2 T \mathcal{G}_1(m,T)\delta + 128(C_3(s+m)+1)\log(T) + 8\Gamma_T,$$

where $\Gamma_T := \max\left\{0, \sum_t \hat{f}_t(\hat{\theta})\right\}$.

Combining Lemma EC.3, (EC.29), $H^* = \left(\sum_i^T \nabla^2 f_{\mathcal{A}_i}(P_0 Q\beta^*)\right)$, $\sum_{t=1}^T \hat{f}_t(\hat{\theta}) = T L_z(\hat{\theta})$ and the definition of ω_t, we can show that

$$\begin{aligned}
&|R_{\mathcal{A}}(\hat{\theta}) - R_{\mathcal{A}}(P_0 Q\beta^*)| - \frac{1}{2}\lambda_{\max}\delta^2 \\
&\le R_{\max}\omega_t \sqrt{\left\| \left(\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(P_0 Q\beta^*) \right)^{-1/2} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) \left(\sum_{i=1}^T \nabla^2 f(P_0 Q\beta^*) \right)^{-1/2} \right\|_{op}}.
\end{aligned}$$

Furthermore, for all $i \le T$, we can show that

$$\begin{aligned}
&\left\| \nabla^2 f_{\mathcal{A}_i}(P_0 Q\beta^*) - \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right\|_{op} \le L_3 \|P_0 Q\beta^* - \hat{\theta}\| = L_3\delta \\
\Rightarrow &\nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) - L_3\delta I \preceq \nabla^2 f_{\mathcal{A}_i}(P_0 Q\beta^*).
\end{aligned}$$

The lower bound for $\nabla^2 f(\hat{\theta})$ can be established by using Lemma EC.6, which shows that the following inequality holds with probability $1 - \mathcal{O}(1/T)$:

$$\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \succeq \frac{1}{2}\mu n_T I.$$

When $\delta \le \frac{\mu n_T}{4 L_3 T}$, we have $\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) - L_3\delta I \succeq \frac{1}{2}\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(\hat{\theta})$, which leads to

$$\sqrt{2}\left(\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-1/2} \succeq \left(\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(P_0 Q\beta^*) \right)^{-1/2}.$$

Therefore, with probability $1 - O(T^{-1})$ the following result holds:

$$|R_{\mathcal{A}}(\hat{\theta}) - R_{\mathcal{A}}(P_0 Q\beta^*)| \le \sqrt{2} R_{\max}\omega_T \sqrt{\left\| \left(\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-1/2} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) \left(\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-1/2} \right\|_{op}} + \frac{1}{2}\lambda_{\max}\delta^2.$$

EC.6. Proof of Corollary 1

If we choose $\mathcal{A}$ to be the assortment with a single item with vector x and $r = 1$, we then have

$$R_{\mathcal{A}}(\theta) = \frac{\exp(x^T Q^T P_0^T \theta)}{1 + \exp(x^T Q^T P_0^T \theta)} \tag{EC.30}$$

Consider the function $\phi(x) = x/(1+x)$, which monotonically increases in x for $x \in (0, x_0)$. Therefore, we have

$$x_1 - x_2 \leq \frac{|\phi(x_1) - \phi(x_2)|}{\phi'(x_0)} \tag{EC.31}$$

Thus, applying Lemma 2, we will have

$$\begin{aligned}
&\exp(x^T\Sigma\beta^*) - \exp(x^TQ^TP_0^T\theta) \leq \frac{|R_s(P_0Q\beta^*) - R_s(\theta)|}{\phi'(\exp(x_{\max}b))} \\
&\leq \frac{\sqrt{2}\omega_t}{\phi'(\exp(x_{\max}b))}\sqrt{\left\|\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2}\nabla^2 f_{\mathcal{A}}(\hat{\theta})\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2}\right\|_{op}} \\
&+ \frac{\lambda_{\max}}{2\phi'(\exp(x_{\max}b))}\delta^2.
\end{aligned} \tag{EC.32}$$

We then bound the difference between $\exp(x^T\beta^*)$ and $\exp(x^T\Sigma\beta^*)$. Via Taylor expansion, there exists a ξ such that

$$\exp(x^T\beta^*) - \exp(x^T\Sigma\beta^*) = \exp(x^T\xi)x^T(\beta^* - \Sigma\beta^*) \leq \exp(x_{\max}b)x_{\max}\mathcal{G}_1(m,T), \tag{EC.33}$$

where the last inequality uses Assumption A.1 and Theorem 2. Combining (EC.32) and eq:tmp3:1, the desirable result follows.

EC.7. Proof of Theorem 3

For simplification, we will ignore the subscript t in this proof. In Corollary 1, we show that for any x we have

$$\exp(x^T\beta^*) \leq v^{ucb}. \tag{EC.34}$$

Let $R^{ucb}_{\Sigma\beta^*}(\mathcal{A}) = \frac{\sum_{i\in\mathcal{A}} r_i v_i^{ucb}}{\sum_{i\in\mathcal{A}} v_i^{ucb}}$. Combining (EC.34) with Lemma A.3 in Agrawal et al. (2019), we can directly show that

$$R^{ucb}_{\Sigma\beta^*}(\mathcal{A}^*) \geq R_{\beta^*}(\mathcal{A}^*).$$

Using the fact that $\mathcal{A}^{SRP}$ is the optimal assortment under utilities $\{v_i^{ucb}\}$, we can further show that

$$R^{ucb}_{\Sigma\beta^*}(\mathcal{A}^{SRP}) \geq R_{\beta^*}(\mathcal{A}^*). \tag{EC.35}$$

In addition, we can show that

$$R^{ucb}_{\Sigma\beta^*}(\mathcal{A}) - R_{\beta^*}(\mathcal{A}) \leq \frac{\sum_{i\in\mathcal{A}} r_i(v_i^{ucb} - \exp(x_i^T\beta^*))}{\sum_{i\in\mathcal{A}} v_i^{ucb}} \leq \sum_{i\in\mathcal{A}} r_i(v_i^{ucb} - \exp(x_i^T\beta^*)), \tag{EC.36}$$

where we use $v^{ucb} \geq \exp(x\beta^*)$ and $\sum_{i\in\mathcal{A}} v_i^{ucb} \geq 1$. Therefore, we can show that

$$\begin{aligned}
R_{\beta^*}(\mathcal{A}^*) - R_{\beta^*}(\mathcal{A}^{SRP}) &\leq R^{ucb}_{\Sigma\beta^*}(\mathcal{A}^{SRP}) - R_{\beta^*}(\mathcal{A}^{SRP}) \\
&\leq R_{\max}\sum_{i\in\mathcal{A}^{SRP}}(v_i^{ucb} - \exp(x_i^T\beta^*)) \\
&= R_{\max}\sum_{i\in\mathcal{A}^{SRP}}\left(\exp(x^TQ^TP_0^T\hat{\theta}) - \exp(x_i^T\beta^*)\right)
\end{aligned}$$

$$+ R_{\max} \sum_{i\in\mathcal{A}^{SRP}} \left(\frac{\lambda_{\max}}{2\phi'(e^{x_{\max}b})}\delta^2 + e^{x_{\max}b} x_{\max}\mathcal{G}_1(m,T) \right)$$
$$+ R_{\max} \sum_{i\in\mathcal{A}^{SRP}} \left(\frac{\sqrt{2}R_{\max}\omega_t}{\phi'(e^{x_{\max}b})} \sqrt{ \left\| \left(\sum_{i=1}^{t} \nabla^2 f_o(\hat{\theta}) \right)^{-1/2} \nabla^2 f_{\mathcal{A}^i}(\hat{\theta}) \left(\sum_{i=1}^{t} \nabla^2 f_{\mathcal{A}_t}(\hat{\theta}) \right)^{-1/2} \right\|_{op} } \right), \tag{EC.37}$$

where the second inequality uses (EC.36). In (EC.37) we denote the assortment with single item $i \in \mathcal{A}$ as $\mathcal{A}^i$ and use the definition of v^{ucb}.

Next, we need to upper bound the term $\sum_{i\in\mathcal{A}^{SRP}} \left(\exp(x_i^T Q^T P_0^T \hat{\theta}) - \exp(x_i^T \beta^*) \right)$. By Taylor expansion, there exists a set of $\{\xi_i : \xi_i \text{ is between } x^T Q^T P_0^T \hat{\theta} \text{ and } x_i^T \beta^*\}$ such that

$$\begin{aligned}
\sum_{i\in\mathcal{A}^{SRP}} \left(\exp(x^T Q^T P_0^T \hat{\theta}) - \exp(x_i^T \beta^*) \right) &= \sum_{i\in\mathcal{A}^{SRP}} \exp(\xi_i) x_i^T (Q^T P_0^T \hat{\theta} - \beta^*) \\
&= \sum_{i\in\mathcal{A}^{SRP}} \exp(\xi_i) x_i^T (Q^T P_0^T \hat{\theta} - \Sigma\beta^* + \Sigma\beta^* - \beta^*) \\
&= \sum_{i\in\mathcal{A}^{SRP}} \exp(\xi_i) \left[x_i^T (Q^T P_0^T \hat{\theta} - \Sigma\beta^*) + x_i^T (\Sigma\beta^* - \beta^*) \right] \\
&= \sum_{i\in\mathcal{A}^{SRP}} \exp(\xi_i) \left[z_i^T (\hat{\theta} - P_0 Q \beta^*) + x_i^T (\Sigma - I)\beta^* \right] \\
&\le \sum_{i\in\mathcal{A}^{SRP}} \exp(\xi_i) \left[\max_i \|z_i\| \delta + x_{\max} \mathcal{G}_1(m,T) \right], \qquad \text{(EC.38)}
\end{aligned}$$

where last inequality uses $\|x_i\| \le x_{\max}$ and $\delta = \|\hat{\theta} - P_0 Q \beta^*\|$ in Assumption A.1 and event $\mathcal{E}_2(m,T)$. Under the event $\mathcal{E}_{rp}(m,d,1/2)$, we have $\|z_i\| \le 2\|x_i\| \le 2x_{\max}$. Combining this result with (EC.38), we have

$$\sum_{i\in\mathcal{A}^{SRP}} \left(\exp(z_i^T \hat{\theta}) - \exp(x_i^T \beta^*) \right) \le \sum_{i\in\mathcal{A}^{SRP}} \exp(\xi_i) \cdot x_{\max}(2\delta + \mathcal{G}_1(m,T)) \le K \exp(x_{\max} b) x_{\max}(2\delta + \mathcal{G}_1(m,T)). \tag{EC.39}$$

Then, we will upper bound the term $\sum_{i\in\mathcal{A}^{SRP}} \sqrt{ \left\| \left(\sum_{i=1}^{T} \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-1/2} \nabla^2 f_{\mathcal{A}^i}(\hat{\theta}) \left(\sum_{i=1}^{T} \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-1/2} \right\|_{op} }$. First, using Lemma EC.8, we can show that

$$\begin{aligned}
\nabla^2 f_{\mathcal{A}}(\hat{\theta}) &= \tilde{\mathbb{E}} \left[\left(z - \tilde{\mathbb{E}}[z] \right) \left(z - \tilde{\mathbb{E}}[z] \right)^T \right] \\
&= \tilde{\mathbb{E}} \left[z z^T \right] - \tilde{\mathbb{E}}[z] \tilde{\mathbb{E}} \left[z^T \right] \\
&= \tilde{\mathbb{E}} \left[z \left(z^T - \tilde{\mathbb{E}}[z] \right)^T \right] \\
&= \sum_{i\in\mathcal{A}} p_{Q^T P_0^T \hat{\theta}}(i) \left[z_i \left(\sum_{j\in\mathcal{A}_t} p_{Q^T P_0^T \hat{\theta}}(j)(z_i - z_j) \right)^T \right] \\
&= \sum_{i,j\in\mathcal{A}} p_{Q^T P_0^T \hat{\theta}}(i) p_{Q^T P_0^T \hat{\theta}}(j) \left[z_i (z_i - z_j)^T \right] \\
&= \sum_{i>j\in\mathcal{A}} p_{Q^T P_0^T \hat{\theta}}(i) p_{Q^T P_0^T \hat{\theta}}(j) \left[z_i (z_i - z_j)^T + z_j (z_j - z_i)^T \right] \\
&= \sum_{i>j\in\mathcal{A}} p_{Q^T P_0^T \hat{\theta}}(i) p_{Q^T P_0^T \hat{\theta}}(j) \left[(z_i - z_j) (z_i - z_j)^T \right]
\end{aligned}$$

$$\succeq \sum_{i\in\mathcal{A}} p_{Q^T P_0^T \hat{\theta}}(i) p_{Q^T P_0^T \hat{\theta}}(0) \left[z_i z_i^T\right]$$

Moreover, by the definition of single item assortment $\mathcal{A}^i$, we have

$$\begin{aligned}
\nabla^2 f_{\mathcal{A}^i}(\hat{\theta}) &= \tilde{\mathbb{E}}\left[\left(z-\tilde{\mathbb{E}}[z]\right)\left(z-\tilde{\mathbb{E}}[z]\right)^T\right] \\
&= p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(i) z_i z_i^T - p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(i)\cdot(z_i)\left(p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(i)\cdot(z_i)\right)^T \\
&= p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(i) p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(0) z_i z_i^T \\
\Rightarrow z_i z_i^T &= \frac{\nabla^2 f_{\mathcal{A}^i}(\hat{\theta})}{p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(i) p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(0)}
\end{aligned}$$

Therefore

$$\begin{aligned}
\nabla^2 f_{\mathcal{A}}(\hat{\theta}) &\succeq \sum_{i\in\mathcal{A}} p_{Q^T P_0^T \hat{\theta}}(i) p_{Q^T P_0^T \hat{\theta}}(0) \left[z_i z_i^T\right] \\
&= \sum_{i\in\mathcal{A}} \frac{p_{Q^T P_0^T \hat{\theta}}(i) p_{Q^T P_0^T \hat{\theta}}(0)}{p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(i) p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(0)} \nabla^2 f_{\mathcal{A}^i}(\hat{\theta}) \\
\Rightarrow \nabla^2 f_{\mathcal{A}}(\hat{\theta}) &\succeq \min_i \left\{ \frac{p_{Q^T P_0^T \hat{\theta}}(i) p_{Q^T P_0^T \hat{\theta}}(0)}{p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(i) p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(0)} \right\} \sum_{i\in\mathcal{A}} \nabla^2 f_{\mathcal{A}^i}(\hat{\theta}),
\end{aligned}$$

which implies that

$$\sum_{i\in\mathcal{A}} \nabla^2 f_{\mathcal{A}^i}(\hat{\theta}) \preceq \max_i \left\{ \frac{p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(i) p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(0)}{p_{Q^T P_0^T \hat{\theta}}(i) p_{Q^T P_0^T \hat{\theta}}(0)} \right\} \nabla^2 f_{\mathcal{A}}(\hat{\theta}).$$

As x and β have upper bounds $x_{\max}$, b and $\Sigma\hat{\beta}$ is feasible, we know that

$$\exp(x^T \Sigma\hat{\beta}) \in [1/\exp(x_{\max} b), \exp(x_{\max} b)].$$

Denote $\eta := \exp(x_{\max} b)$ for shorthand. Then

$$\begin{aligned}
&p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(i) \le \eta/(1+\eta),\ p_{Q^T P_0^T \hat{\theta},\mathcal{A}^i}(0) \le \eta/(1+\eta) \\
&p_{Q^T P_0^T \hat{\theta}}(i) \ge (\eta + K\eta^2)^{-1},\ p_{Q^T P_0^T \hat{\theta}}(0) \ge (1+K\eta)^{-1}
\end{aligned}$$

Thus we have

$$\begin{aligned}
&\sum_i \nabla^2 f_{\mathcal{A}^i}(\hat{\theta}) \preceq \frac{\eta^2(\eta+K\eta^2)(1+K\eta)}{(1+\eta)^2} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) = \frac{\eta^3(1+K\eta)^2}{(1+\eta)^2} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) \\
\Rightarrow &\nabla^2 f_{\mathcal{A}^i}(\hat{\theta}) \preceq \frac{\eta^3(1+K\eta)^2}{(1+\eta)^2} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) \\
\Rightarrow &\sum_{i\in\mathcal{A}^{SRP}} \sqrt{\left\| \left(\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2} \nabla^2 f_{\mathcal{A}^i}(\hat{\theta}) \left(\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2} \right\|_{op}} \\
&\le K \sqrt{\left\| \left(\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2} \frac{\eta^3(1+K\eta)^2}{(1+\eta)^2} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) \left(\sum_{i=1}^T \nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2} \right\|_{op}}
\end{aligned}$$

$$= \frac{K\eta^{3/2}(1+K\eta)}{(1+\eta)} \sqrt{\left\| \left(\sum_{i=1}^{T} \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-1/2} \nabla^2 f_{\mathcal{A}}(\hat{\theta}) \left(\sum_{i=1}^{T} \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-1/2} \right\|_{op}}. \tag{EC.40}$$

Finally, the theorem follows directly by combining (EC.37), (EC.39), and (EC.40).

EC.8. Proof of Theorem 4

Let both events $\mathcal{E}_{lasso}(T)$ and $\mathcal{E}_{rp}(m,d,1/2)$ hold. We separate the expected cumulative regret under random samples from that without random samples:

$$\text{REGRET}(T) \leq \overbrace{\sum_{t\in random}^{T} \mathbb{E}[R_{t,\beta^*}(\mathcal{A}_t^*) - R_{t,\beta^*}(\mathcal{A}_t)]}^{\text{(Random samples)}} + \overbrace{\sum_{t\notin random}^{T} \mathbb{E}[R_{t,\beta^*}(\mathcal{A}_t^*) - R_{t,\beta^*}(\mathcal{A}_t)]}^{\text{(Non-random samples)}}.$$

Non-random samples part: Recall that we periodically update the projection matrix P_0 based on the Lasso problem. We start with considering the cumulative regret for a arbitrary single period. Without loss of generality, we consider the period starting from T_a and ending at T_b.

We first consider the situation where $\mathcal{G}_0(T) \geq \frac{1}{3}\beta_{\min}$. By Theorem 3, with high probability we have

$$\begin{aligned}
&\sum_{t\notin random, t\in[T_a,T_b)} R_{t,\beta^*}(\mathcal{A}^*) - R_{t,\beta^*}(\mathcal{A}_t) \\
&\leq \overbrace{\sum_{t\notin random, t\in[T_a,T_b)} 2R_{\max}K\eta x_{\max}\mathcal{G}_1(m,t)}^{\text{ⓐ}} \\
&+ \overbrace{\sum_{t\notin random, t\in[T_a,T_b)} \frac{R_{\max}K\lambda_{\max}}{2\phi'(e^{x_{\max}b})}\delta^2 + 2KR_{\max}\eta x_{\max}\delta}^{\text{ⓑ}} \\
&+ \overbrace{\sum_{t\notin random, t\in[T_a,T_b)} \min\left\{ R_{\max}, \frac{K\eta^{3/2}(1+K\eta)\sqrt{2}\omega_t}{\phi'(e^{x_{\max}b})(1+\eta)} \sqrt{\left\| \left(\sum_{i=1}^{T} \nabla^2 f_{\mathcal{A}_i}(\hat{\theta}) \right)^{-1/2} \nabla^2 f_{\mathcal{A}\mathcal{S}\mathcal{R}\mathcal{P}}(\hat{\theta}) \left(\sum_{I=1}^{t} \nabla^2 f(\hat{\theta}) \right)^{-1/2} \right\|_{op}} \right\}}^{\text{ⓒ}}.
\end{aligned} \tag{EC.41}$$

The bound for part ⓐ:

$$\begin{aligned}
\text{ⓐ} &\leq \sum_{t=T_a}^{T_b-1} 2KR_{\max}\eta x_{\max} \cdot \frac{6(C_1^2+\log d)s + \log t(6/C_1+6/C_2) + 6C_2 m}{m} \sqrt{s}\cdot C_{lasso} s \sqrt{\frac{\log d + \log t}{n_{T_a}}} \\
&\leq \sum_{t=T_a}^{T_b-1} 2KR_{\max}\eta x_{\max} \cdot \frac{6(C_1^2+\log d)s + \log T(6/C_1+6/C_2) + 6C_2 m}{m} \sqrt{s}\cdot C_{lasso} s \sqrt{\frac{\log d + \log T}{n_{T_a}}} \\
&\lesssim \mathcal{O}\left(s^{\frac{5}{2}} \log^{\frac{3}{2}} d \log^{\frac{3}{2}} T m (T_b - T_a - 1) n_{T_a}^{-\frac{1}{2}} \right).
\end{aligned} \tag{EC.42}$$

The bound for part ⓑ: As $\delta_t \leq \frac{n_t\mu}{4tL_3}$, we can show that

$$\text{ⓑ} \leq \frac{KR_{\max}\lambda_{\max}}{2\phi'(e^{x_{\max}b})} \sum_{t=T_a}^{T_b-1} \delta^2 + 2KR_{\max}\eta x_{\max} \sum_{t=T_a}^{T_b-1} \delta$$

$$\leq \frac{\mu^2 K R_{\max}\lambda_{\max}}{32L_3^2\phi'(e^{(x_{\max}b)})}\sum_{t=T_a}^{T_b-1}\frac{n_t^2}{t^2}+\frac{\mu K R_{\max}\eta x_{\max}}{2L_3}\sum_{t=T_a}^{T_b-1}\frac{n_t}{t}$$
$$=\mathcal{O}(\sum_{t=T_a}^{T_b-1} n_t^2/t^2)+\mathcal{O}(\sum_{t=T_a}^{T_b-1} n_t/t)=\mathcal{O}(\sum_{t=T_a}^{T_b-1} n_t/t).$$

The bound for part ©:

Note for $R_{\max}\geq 1$ and $\frac{KR^2\eta^{3/2}(1+K\eta)\sqrt{2}\omega_t}{\phi'(e^{x_{\max}b})(1+\eta)}\geq 1$, we can show that

$$\text{©}\leq \frac{R_{\max}^3 K\eta^{3/2}(1+K\eta)\sqrt{2}\omega_t}{\phi'(e^{x_{\max}b})(1+\eta)}\sum_{t=T_a}^{T_b-1}\min\left\{1,\sqrt{\left\|\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2}\nabla^2 f_{\mathcal{A}_t}(\hat{\theta})\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2}\right\|_{op}}\right\}$$
$$\leq \frac{R_{\max}^3 K\eta^{3/2}(1+K\eta)\sqrt{2}\omega_t}{\phi'(e^{x_{\max}b})(1+\eta)}\sqrt{T_b-T_a-1}\sqrt{\sum_{t=T_a}^{T_b-1}\min\left\{1,\left\|\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2}\nabla^2 f_{\mathcal{A}_t}(\hat{\theta})\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2}\right\|_{op}\right\}},$$

where the last inequality uses the fact that $\sum_{i=1}^b\sqrt{c_i}\leq\sqrt{b}\sqrt{\sum_{i=1}^b c_i}$ holds for all $b,c_i>0$. As $f(\cdot)$ has Lipschitz hessian, we have

$$\left\|\nabla^2 f_{\mathcal{A}_t}(\hat{\theta})-\nabla^2 f_{\mathcal{A}_t}(P_0Q\beta^*)\right\|_{op}\leq L_3\|\hat{\theta}-P_0Q\beta^*\|\leq L_3\delta_t. \tag{EC.43}$$

When $\delta_t\leq\frac{n_t\mu}{4tL_3}$, using Lemma EC.6, we can verify that with high probability

$$\sum_t\nabla^2 f_{\mathcal{A}_t}(\hat{\theta})\succeq\frac{1}{2}\sum_t\nabla^2 f_{\mathcal{A}_t}(P_0Q\beta^*)$$
$$\Rightarrow\left(\sum_t\nabla^2 f_{\mathcal{A}_t}(\hat{\theta})\right)^{-1/2}\preceq\sqrt{2}\left(\sum_t\nabla^2 f_{\mathcal{A}_t}(P_0Q\beta^*)\right)^{-1/2}.$$

Therefore,

$$\sum_{t=T_a}^{T_b-1}\min\left\{1,\left\|\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2}\nabla^2 f_{\mathcal{A}_t}(\hat{\theta})\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(\hat{\theta})\right)^{-1/2}\right\|_{op}\right\}$$
$$\leq 2\sum_{t=T_a}^{T_b-1}\min\left\{1,\left\|\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(P_0Q\beta^*)\right)^{-1/2}\nabla^2 f_{\mathcal{A}_t}(\hat{\theta})\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(P_0Q\beta^*)\right)^{-1/2}\right\|_{op}\right\}$$
$$\leq 2\sum_{t=T_a}^{T_b-1}\min\left\{1,\left\|\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(P_0Q\beta^*)\right)^{-1/2}\nabla^2 f_{\mathcal{A}_t}(P_0Q\beta^*)\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(P_0Q\beta^*)\right)^{-1/2}\right\|_{op}\right\}$$
$$+2\sum_{t=T_a}^{T_b-1}\left\|\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(P_0Q\beta^*)\right)^{-1/2}\left(\nabla^2 f_{\mathcal{A}_t}(\hat{\theta})-\nabla^2 f_{\mathcal{A}_t}(P_0Q\beta^*)\right)\left(\sum_{i=1}^{t-1}\nabla^2 f_{\mathcal{A}_i}(P_0Q\beta^*)\right)^{-1/2}\right\|_{op}$$
$$\leq 4(s+m)\log\left(\frac{8(T_b-1)K^2x_{\max}^2}{\mu n_{T_a}}\right)+2\sum_{t=T_a}^{T_b-1}L_3\delta_t\left\|\left(\sum_t\nabla^2 f(P_0Q\beta^*)\right)^{-1}\right\|_{op} \tag{EC.44}$$
$$\leq 4(s+m)\log\left(\frac{8(T_b-1)K^2x_{\max}^2}{\mu}\right)+\sum_{t=T_a}^{T_b-1}\frac{4L_3\delta_t}{\mu n_t} \tag{EC.45}$$

$$\leq 4(s+m)\log\left(\frac{8(T_b-1)K^2x_{\max}^2}{\mu}\right)+\sum_{t=T_a}^{T_b-1}\frac{4L_3}{\mu n_t}\frac{n_t\mu}{4tL_3} \tag{EC.46}$$

$$\lesssim \mathcal{O}\left((s+m)\log(T_b-1)\sum_{t=T_a}^{T_b-1}t^{-1}\right)=\mathcal{O}\left((s+m)\log^2(T_b-1)\right),$$

where (EC.44) uses Lemma EC.7 and (EC.43), (EC.45) uses Lemma EC.6, and (EC.46) uses $\delta_t \leq \frac{n_t\mu}{4tL_3}$. Therefore, we can upper bound part ⓒ as follow:

$$ⓒ \lesssim \mathcal{O}\left(\omega_{T_b}\sqrt{T_b-T_a-1}(s+m)\log^2(T_b-1)\right)$$

As $\omega_{T_b}=4\sqrt{4x_{\max}^2T_b\mathcal{G}_1(m,T_b)\delta+32C_3(s+m))\log(T_b)+2\Gamma_{T_b}}$, $\mathcal{G}_1(m,T_b)=\mathcal{O}(s^{\frac{5}{2}}\log^{\frac{3}{2}}d\log^{\frac{3}{2}}T_bmn_{T_b}^{-\frac{1}{2}})$, and $\delta_{T_b}=\mathcal{O}(n_{T_b}/T_b)$. We can verify that $\omega_{T_b}\lesssim\mathcal{O}(s^{\frac{5}{4}}m^{\frac{1}{2}}\log^{\frac{3}{4}}d\log^{\frac{3}{4}}T_bn_{T_b}^{\frac{1}{4}})$. Directly, we have

$$ⓒ \lesssim \mathcal{O}\left(s^{\frac{9}{4}}m^{\frac{3}{2}}\log^{\frac{3}{4}}d\log^{\frac{11}{4}}T_bn_{T_b}^{\frac{1}{4}}\sqrt{T_b-T_a-1}\right).$$

Hence, when $\mathcal{G}_0(T)\geq\frac{1}{3}\beta_{\min}$, we can show that the upper bound for non-random part is

$$\sum_{t\notin random,t\in[T_a,T_b)}^{T}\mathbb{E}[R_{t,\beta^*}(\mathcal{A}_t^*)-R_{t,\beta^*}(\mathcal{A}_t)]$$

$$\lesssim \mathcal{O}\left(s^{\frac{5}{2}}\log d\log^{\frac{3}{2}}Tm(T_b-T_a-1)n_{T_a}^{-\frac{1}{2}}+\sum_{t=T_a}^{T_b-1}n_t/t+s^{\frac{9}{4}}m^{\frac{3}{2}}\log^{\frac{3}{4}}d\log^{\frac{11}{4}}T_bn_{T_b}^{\frac{1}{4}}\sqrt{T_b-T_a-1}\right). \tag{EC.47}$$

By Lemma EC.9, we know that with high probability, when $\mathcal{G}_0(T)\geq\frac{1}{3}\beta_{\min}$, we have

$$n_t=\mathcal{O}(C_0t^{2/3})=\mathcal{O}(s^2\log d\log Tt^{2/3}) \tag{EC.48}$$

Combining (EC.48) and (EC.47), we can show that

$$\sum_{t\notin random,t\in[T_a,T_b)}^{T}\mathbb{E}[R_{t,\beta^*}(\mathcal{A}_t^*)-R_{t,\beta^*}(\mathcal{A}_t)]$$

$$\lesssim \mathcal{O}\left(s^{\frac{3}{2}}\log d^{\frac{1}{2}}\log Tm(T_b-T_a-1)T_a^{-\frac{1}{3}}+s^2\log d\log T\sum_{t=T_a}^{T_b-1}t^{-1/3}+s^{\frac{11}{4}}m^{\frac{3}{2}}\log d\log^3T_bT_b^{\frac{1}{6}}\sqrt{T_b-T_a-1}\right)$$

$$\lesssim \mathcal{O}\left(s^{\frac{11}{4}}m^{\frac{3}{2}}\log d\log^3T_b\left((T_b-T_a-1)T_a^{-\frac{1}{3}}+T_b^{\frac{2}{3}}-T_a^{\frac{2}{3}}+T_b^{\frac{1}{6}}\sqrt{T_b-T_a-1}\right)\right). \tag{EC.49}$$

As we use $T_{lasso}=\{c^i,i=0,1,2,...\}$ random sampling schedule, we can show that

$$T_b-T_a-1=c^i-c^{i-1}-1=c^{i-1}(c-1)-1,\ i=1,2,...$$

Hence, we can further simplify (EC.49) as follows:

$$\sum_{t\notin random,t\in\text{Period }i-1}\mathbb{E}[R_{t,\beta^*}(\mathcal{A}_t^*)-R_{t,\beta^*}(\mathcal{A}_t)]$$

$$\lesssim \mathcal{O}\left(s^{\frac{11}{4}}m^{\frac{3}{2}}\log d\log^3T_b\left((c^i-c^{i-1})c^{-(i-1)/3}+c^{2i/3}-c^{2(i-1)/3}+c^{i/6t}(c^i-c^{i-1})^{\frac{1}{2}}\right)\right)$$

$$=\mathcal{O}\left(s^{\frac{11}{4}}m^{\frac{3}{2}}\log d\log^3T_bc^{2i/3}\right)=\mathcal{O}\left(s^{\frac{11}{4}}m^{\frac{3}{2}}\log d\log^3T_bT_b^{2/3}\right). \tag{EC.50}$$

Using (EC.50) and the fact that up to time T we will have $\mathcal{O}(\log T)$ number of epochs, we can show that

$$\sum_{t\notin random} \mathbb{E}[R_{t,\beta^*}(\mathcal{A}_t^*) - R_{t,\beta^*}(\mathcal{A}_t)] \lesssim \mathcal{O}\left(s^{\frac{11}{4}} m^{\frac{3}{2}} \log d \log^4 T \cdot T^{2/3}\right). \tag{EC.51}$$

Random samples part: Since we use random decay sampling schedule, we have

$$\sum_{t\in random}^{T} \mathbb{E}[R_{t,\beta^*}(\mathcal{A}_t^*) - R_{t,\beta^*}(\mathcal{A}_t)] \leq R_{\max} n_T \lesssim \mathcal{O}(n_T) = \mathcal{O}(s^2 \log d \log T \cdot T^{\frac{2}{3}}). \tag{EC.52}$$

Combining both non-random samples part and random samples part, i.e., (EC.52) and (EC.51), we can upper bound the cumulative regret up to time T for the case when $\mathcal{G}_0(T) \geq \frac{1}{3}\beta_{\min}$ as follows:

$$\sum_{t} \mathbb{E}[R_{t,\beta^*}(\mathcal{A}_t^*) - R_{t,\beta^*}(\mathcal{A}_t)] \lesssim \mathcal{O}\left(s^{\frac{11}{4}} m^{\frac{3}{2}} \log d \log^4 T \cdot T^{2/3}\right). \tag{EC.53}$$

Following the similar procedure, we can show when $\mathcal{G}_0(T) < \frac{1}{3}\beta_{\min}$,

$$\sum_{t} \mathbb{E}[R_{t,\beta^*}(\mathcal{A}_t^*) - R_{t,\beta^*}(\mathcal{A}_t)] \lesssim \mathcal{O}\left(s^2 m \log d \log^3 T \cdot T^{\frac{1}{2}}\right). \tag{EC.54}$$

Note that in previous proofs, we assume events $\mathcal{E}_{lasso}(T)$ and $\mathcal{E}_{rp}(m, d, 1/2)$ hold. The remaining task is to bound the probability that those two events happen simultaneously. As we require $C_0 = \mathcal{O}(s^2 \log d \log T)$, then there exists a T_0 such that $n_T \geq \mathcal{O}(s^2 \log T)$ for $T \geq T_0$. Using Lemma 1, we have

$$\mathbb{P}(\mathcal{E}_{lasso}(T)) \geq 1 - \mathcal{O}(T^{-1}). \tag{EC.55}$$

Via Lemma 2, we have

$$\mathbb{P}(\mathcal{E}_{rp}(m, d, 1/2)) \geq 1 - 2\exp(-C_2 m). \tag{EC.56}$$

We then use the union bound over (EC.55) and (EC.56) and the desirable result follows:

$$\mathbb{P}(\mathcal{E}_{lasso}(T) \cap \mathcal{E}_{rp}(m, d, 1/2)) \geq 1 - 2\exp(-C_2 m) - \mathcal{O}(T^{-1}).$$

EC.9. Technical Lemmas

EC.9.1. Proof of Lemma EC.1

To simplify the notation in this proof, we use $\nabla^2 L(\xi)$ to denote $\nabla^2 L(\xi|x,\mathcal{A})$, which can be re-written as follows:

$$\begin{aligned}
\nabla^2 L(\xi) &= -\frac{1}{n}\sum_{i=1}^{n}\left(\frac{(\sum_{k\in\mathcal{A}_i}\exp(x_{k,i}^T\xi)x_{k,i})(\sum_{k\in\mathcal{A}_i}\exp(x_{k,i}^T\xi)x_{k,i}^T)}{(\sum_{k\in\mathcal{A}_i}\exp(x_{k,i}^T\xi)^2} - \frac{\sum_{k\in\mathcal{A}_i}\exp(x_{k,i}^T\xi)x_{k,i}x_{k,i}^T}{\sum_{k\in\mathcal{A}_i}\exp(x_{k,i}^T\xi)}\right)\\
&= -\frac{1}{n}\sum_{i=1}^{n}\frac{\sum_{k_1\in\mathcal{A}_i}\sum_{k_2\in\mathcal{A}_i}\exp(x_{k_1,i}^T\xi)\exp(x_{k_2,i}^T\xi)x_{k_1,i}x_{k_2,i}^T}{(\sum_{k\in\mathcal{A}_i}\exp(x_{k,i}^T\xi)^2}\\
&+ \frac{1}{n}\sum_{i=1}^{n}\frac{\sum_{k_1\in\mathcal{A}_i}\sum_{k_2\in\mathcal{A}_i}\exp(x_{k_2,i}^T\xi)\exp(x_{k_1,i}^T\xi)x_{k_1,i}x_{k_1,i}^T}{(\sum_{k\in\mathcal{A}_i}\exp(x_{k,i}^T\xi))^2}\\
&= -\frac{1}{n}\sum_{i=1}^{n}\sum_{k1,k2}\frac{\exp(x_{k_2,i}^T\xi)\exp(x_{k_1,i}^T\xi)x_{k_1,i}(x_{k_2,i}-x_{k_1,i})^T}{(\sum_{k\in\mathcal{A}_i}\exp(x_{k,i}^T\xi))^2}\\
&= \frac{1}{n}\sum_{i=1}^{n}\sum_{k_1,k_2}\Phi_i(k_1,k_2),
\end{aligned}$$

where we use $\Phi_i(k_1,k_2)$ to denote the following shorthand:

$$\Phi_i(k_1,k_2) := -\frac{\exp(x_{k_2,i}^T\xi)\exp(x_{k_1,i}^T\xi)x_{k_1,i}(x_{k_2,i}-x_{k_1,i})^T}{(\sum_{k\in\mathcal{A}_i}\exp(x_{k,i}^T\xi))^2}$$

Further, we denote $\phi_i(k_1,k_2) := \frac{\exp(x_{k_2,i}^T\xi)\exp(x_{k_1,i}^T\xi)}{(\sum_{k\in\mathcal{A}_i}\exp(x_{k,i}^T\xi))^2}$. Hence, when $k_1\neq k_2$, we have

$$\begin{aligned}
\Phi_i(k_1,k_2)+\Phi_i(k_2,k_1) &= -\frac{\exp(x_{k_2,i}^T\xi)\exp(x_{k_1,i}^T\xi)(x_{k_1,i}(x_{k_2,i}-x_{k_1,i})^T + x_{k_2,i}(x_{k_1,i}-x_{k_2,i})^T)}{(\sum_{k\in\mathcal{A}_i}\exp(x_{k,i}^T\xi))^2}\\
&= -\phi_i(k_1,k_2)(x_{k_1,i}(x_{k_2,i}-x_{k_1,i})^T + x_{k_2,i}(x_{k_1,i}-x_{k_2,i})^T),
\end{aligned}$$

Then, for any $z\in\mathbb{R}^d$, we have

$$\begin{aligned}
\frac{z^T(\Phi_i(k_1,k_2)+\Phi_i(k_2,k_1))z}{\phi_i(k_1,k_2)} &= -z^T(x_{k_1,i}(x_{k_2,i}-x_{k_1,i})^T + x_{k_2,i}(x_{k_1,i}-x_{k_2,i})^T)z\\
&= -\left(z^Tx_{k_1,i}x_{k_2,i}^Tz - z^Tx_{k_1,i}x_{k_1,i}^Tz + z^Tx_{k_2,i}x_{k_1,i}^Tz - z^Tx_{k_2,i}x_{k_2,i}^Tz\right)\\
&= -\left(-\|z^Tx_{k_1,i}\|^2 - \|z^Tx_{k_2,i}\|^2 + 2\langle z^Tx_{k_1,i}, z^Tx_{k_2,i}\rangle\right)\\
&= \|z^T(x_{k_1,i}-x_{k_2,i})\|^2 \geq 0
\end{aligned}$$

Further, we can show that

$$\begin{aligned}
z^T\nabla^2 L(\xi)z &= \frac{1}{n}\sum_i\sum_{k_1,k_2} z^T\Phi_i(k_1,k_2)z\\
&= \frac{1}{n}\sum_i\sum_{k_1<k_2}\phi_i(k_1,k_2)\|z^T(x_{k_1,i}-x_{k_2,i})\|^2\\
\Rightarrow \nabla^2 L(\xi) &= \frac{1}{n}\sum_i\sum_{k_1<k_2}\phi_i(k_1,k_2)(x_{k_2,i}-x_{k_2,i})(x_{k_2,i}-x_{k_2,i})^T\\
\Rightarrow \nabla^2 L(\xi) &= \frac{1}{n}\sum_i\sum_{k_1<k_2} y_{k_1,k_2,i}y_{k_1,k_2,i}^T, \qquad \text{(EC.57)}
\end{aligned}$$

where we set $y_{k_1,k_2,i} = \sqrt{\phi_i(k_1,k_2)}(x_{k_1,i} - x_{k_2,1})$. By the definition of $\phi_i(k_1,k_2)$, we know that $\|y_{k_1,k_2,i}\|_\infty \le 2x_{\max} := y_{\max}$. Let $K = y_{\max}$, $\sigma_0 = \sqrt{2}y_{\max}$, we can verify the follow inequalitys hold

$$K^2\left(\mathbb{E}[\exp(y_{k1,k2,i,j}^2/K^2) - 1]\right) \le y_{\max}^2(e-1) \le \sigma_0^2, \tag{EC.58}$$

where $y_{k1,k2,i,j}$ is the j-th element of $y_{k1,k2,i}$. Then, via the exercise 14.3 in Bühlmann and Van De Geer (2011), we have the following inequality for $t > 0$:

$$\begin{aligned}&\mathbb{P}\left\{\left\|\frac{2}{nK(K-1)}\sum_i\sum_{k_1<k_2} y_{k_1,k_2,i}y_{k_1,k_2,i}^T - \mathbb{E}[y_{k_1,k_2,i}y_{k_1,k_2,i}^T]\right\|_\infty\right.\\ &\left.\ge 2y_{\max}^2 t + 4z_{\max}^2\sqrt{t} + \sqrt{8}y_{\max}^2\lambda\left(\frac{\sqrt{2}}{2}, n, \binom{d}{2}\right)\right\} \le \exp\left(-\frac{1}{2}nK(K-1)t\right),\end{aligned} \tag{EC.59}$$

where

$$\lambda\left(\frac{\sqrt{2}}{2}, n, \binom{d}{2}\right) = \sqrt{\frac{2\log(d(d-1))}{n}} + \frac{y_{\max}\log(d(d-1))}{n}$$

Note that when $t < 1$ and $n \ge \log d/t$, we will have the following inequalities:

$$2y_{\max}^2 t + 4y_{\max}^2\sqrt{t} \le 6y_{\max}^2\sqrt{t}$$

$$\sqrt{8}y_{\max}^2\lambda\left(\frac{\sqrt{2}}{2}, n, \binom{d}{2}\right) \le \sqrt{8}y_{\max}^2\left(\sqrt{\frac{4\log d}{n}} + \frac{2y_{\max}\log d}{n}\right) \le 4\sqrt{2}y_{\max}^2(1+y_{\max})\sqrt{t}.$$

Combining these two inequalities, we have

$$2y_{\max}^2 t + 4y_{\max}^2\sqrt{t} + \sqrt{8}y_{\max}^2\lambda\left(\frac{\sqrt{2}}{2}, n, \binom{d}{2}\right) \le 2y_{\max}^2(3 + 2\sqrt{2}(1+y_{\max}))\sqrt{t}. \tag{EC.60}$$

When $t = \left(\frac{\kappa}{64sy_{\max}^2(3+2\sqrt{2}(1+y_{\max}))}\right)^2 = \left(\kappa/256sx_{\max}^2(3+2\sqrt{2}(1+2x_{\max})\right)^2$, we have

$$2y_{\max}^2(3 + 2\sqrt{2}(1+y_{\max}))\sqrt{t} = \frac{\kappa}{32s}. \tag{EC.61}$$

Via (EC.59), (EC.60) and (EC.61), with probability $1 - \exp(\frac{1}{2}nK(K-1)t)$ we have

$$\left\|\frac{2}{nK(K-1)}\sum_i\sum_{k_1<k_2} y_{k_1,k_2,i}y_{k_1,k_2,i}^T - \mathbb{E}[y_{k_1,k_2,i}y_{k_1,k_2,i}^T]\right\|_\infty \le \frac{\kappa}{23s}. \tag{EC.62}$$

Then, via the Corollary 6.8 in Bühlmann and Van De Geer (2011), when Assumption A.2 holds, then (EC.62) leads the following result

$$\begin{aligned}\|u_S\|_1^2 &\le \frac{s}{\kappa/2}u^T\left[\frac{2}{nK(K-1)}\sum_i\sum_{k_1<k_2} y_{k_1,k_2,i}y_{k_1,k_2,i}^T\right]u\\ \Rightarrow \|u_S\|_1^2 &\le \frac{4s}{K(K-1)\kappa}u^T\nabla^2 L(\xi)u,\end{aligned} \tag{EC.63}$$

where last inequality uses (EC.57).

Hence, when choosing $C = \frac{1}{2}K(K-1)\left(\kappa/256sx_{\max}^2(3+2\sqrt{2}(1+2x_{\max})\right)^2$, then with probability $1 - \exp(-Cn)$, we have

$$u^T\nabla^2(\xi|x,\mathcal{A})u \ge \frac{K(K-1)\kappa}{4s}\|u_S\|_1^2,$$

where $\|u_{S^C}\|_1 \le 3\|u_S\|$. The remaining of this lemma follows directly by using $n > \log T/C$.

EC.9.2. Proof of Lemma EC.2

We start with showing that the expectation of $\nabla \log(p_{\beta^*,\mathcal{A}}(j))$ among all possible choice $j \in \mathcal{A}$ is 0, i.e., $\mathbb{E}_{j\in\mathcal{A}}[\nabla \log(p_{\beta^*,\mathcal{A}}(j))] = 0$.

$$\begin{aligned}
\mathbb{E}_{j\mathcal{A}}[\nabla \log(p_{\beta^*,\mathcal{A}}(j))] &= \frac{1}{n}\sum_{j\in\mathcal{A}_t} p_{\beta^*,\mathcal{A}}(j)\nabla \log(p_{\beta^*,\mathcal{A}}(j)) \\
&= \sum_{j\in\mathcal{A}_t} p_{\beta^*,\mathcal{A}}(j)\cdot \frac{1}{p_{\beta^*,\mathcal{A}}(j)}\nabla p_{\beta^*,\mathcal{A}}(j) \\
&= \sum_{j\in\mathcal{A}_t} \nabla p_{\beta^*,\mathcal{A}}(j) \\
&= \sum_{j\in\mathcal{A}_t} \nabla \left(\frac{\exp(x_j^T\beta^*)}{\sum_{i\in\mathcal{A}_t}\exp(x_i^T\beta^*)}\right) \\
&= \sum_{j\in\mathcal{A}_t} \left(\frac{\exp(x_j^T\beta^*)x_j}{\sum_{i\in\mathcal{A}_t}\exp(x_i^T\beta^*)} - \frac{\exp(x_j^T\beta^*)\sum_{i\in\mathcal{A}_t}\exp(x_i^T\beta^*)x_i}{(\sum_{i\in\mathcal{A}_t}\exp(x_i^T\beta^*))^2}\right) \\
&= \frac{\sum_{j\in\mathcal{A}_t}\exp(x_j^T\beta^*)x_j}{\sum_{i\in\mathcal{A}_t}\exp(x_i^T\beta^*)} - \frac{\sum_{j\in\mathcal{A}_t}\exp(x_j^T\beta^*)\cdot\sum_{i\in\mathcal{A}_t}\exp(x_i^T\beta^*)x_i}{(\sum_{i\in\mathcal{A}_t}\exp(x_i^T\beta^*))^2} \\
&= \frac{\sum_{j\in\mathcal{A}_t}\exp(x_j^T\beta^*)x_j}{\sum_{i\in\mathcal{A}_t}\exp(x_i^T\beta^*)} - \frac{\sum_{i\in\mathcal{A}_t}\exp(x_i^T\beta^*)x_i}{\sum_{i\in\mathcal{A}_t}\exp(x_i^T\beta^*)} \\
&= 0
\end{aligned}$$

Combining with the fact that $\nabla \log(p_{\beta^*,\mathcal{A}}(j))$ is element-wise bounded by $x_{\max}$, we can conclude that every dimension of $\nabla \log(p_{\beta^*,\mathcal{A}}(j))$ is a zero mean $x_{\max}^2$-subgaussian random variable and that $\nabla L(\beta)$ is the finite average of zero mean i.i.d subgaussian random vector, i.e.,$\nabla L(\beta^*) = \frac{1}{n}\sum_t \nabla \log(p_{\beta^*,\mathcal{A}}(c_t))$.

Via Hoeffding inequality, for any $\epsilon > 0$, we have

$$\mathbb{P}\left(\left|\sum_t \nabla_i \log(p_{\beta^*,\mathcal{A}}(c_t))\right| \ge \epsilon\right) \le 2\exp\left(-\frac{\epsilon^2}{2nx_{\max}^2}\right), \tag{EC.64}$$

where $\nabla_i \log(q_{\beta^*}(c_t))$ is the i-th dimension of $\nabla \log(q_{\beta^*}(c_t))$. Hence, via union bound, we have

$$\begin{aligned}
&\mathbb{P}\left(\left\|\sum_t \nabla \log(p_{\beta^*,\mathcal{A}}(c_t))\right\|_\infty \ge \epsilon\right) \le 2d\exp\left(-\frac{\epsilon^2}{2nx_{\max}^2}\right) \\
\Rightarrow &\mathbb{P}\left(\|\nabla L(\beta^*|x,\mathcal{A})\|_\infty \ge \epsilon/n\right) \le 2\exp\left(-\frac{\epsilon^2}{2nx_{\max}^2} + \log(d)\right).
\end{aligned}$$

If we set $\epsilon = \sqrt{2nx_{\max}^2(\log d + \log T)}$, then

$$\begin{aligned}
&\Rightarrow \mathbb{P}\left(\|\nabla L(\beta^*|x,\mathcal{A})\|_\infty \ge \frac{\sqrt{2nx_{\max}^2(\log d + \log T)}}{n}\right) \le \frac{2}{T} \\
&\Rightarrow \mathbb{P}\left(\|\nabla L(\beta^*|x,\mathcal{A})\|_\infty \ge \sqrt{\frac{2x_{\max}^2(\log d + \log T)}{n}}\right) \le \frac{2}{T}.
\end{aligned}$$

EC.9.3. Proof of Lemma EC.3

By standard covering number arguments (e.g., van de Geer, 2000), an ϵ covering set $\mathcal{H}(\epsilon)$ for $\|\theta - \theta_0\| \le \delta$ has a finite elements upper bounded by $\exp(C_3(s+m)\log(\delta/\epsilon))$, where C_3 is a positive constant. As we require

event $\mathcal{E}_2(m,T)$, $\delta \le \frac{\rho}{8Kx_{\max}}$ and $\mathcal{G}_1(m,T) \le \frac{\rho}{8Kx_{\max}}$, via Lemma EC.4 and the union bound, any $\theta \in \mathcal{H}(\epsilon)$, we can show that the following inequality holds with probability $1-\delta_4 \cdot \exp(C_3(s+m)\log(\delta/\epsilon))$:

$$\left|\sum_{t=1}^{T}[\hat{f}_t(\theta) - f_{\mathcal{A}_t}(\theta)]\right| \le \frac{2}{3}\log(1/\delta_4) + 3\sqrt{\log(1/\delta_4)\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta)} \tag{EC.65}$$

If we set $\delta_4 = \exp(-C_3(s+m)\log(\delta/\epsilon) - \log T)$ and $\epsilon = \frac{1}{2}\delta$, the above inequality directly suggests that the following result holds with probability $1-T^{-1}$:

$$\begin{aligned}\left|\sum_{t=1}^{T}[\hat{f}_t(\theta) - f_{\mathcal{A}_t}(\theta)]\right| &\le \frac{2}{3}(C_3(s+m)\log(2)+\log(T)) + 3\sqrt{(C_3(s+m)\log(2)+\log(T))\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta)} \\ &\le \frac{2}{3}(C_3(s+m)+1)\log(T) + 3\sqrt{(C_3(s+m)+1)\log(T)\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta)},\end{aligned} \tag{EC.66}$$

where last inequality we uses $T \ge 2$ in Lemma statement.

Next, we will bound the term $|\sum_t f_{\mathcal{A}_t}(\hat{\theta})|$:

$$\begin{aligned}|\sum_t f_{\mathcal{A}_t}(\hat{\theta})| &= \sum_t f_{\mathcal{A}_t}(\hat{\theta}) \\ &= \sum_t f_{\mathcal{A}_t}(\hat{\theta}) - \sum_t \hat{f}_t(\hat{\theta}) + \sum_t \hat{f}_t(\hat{\theta}) \\ &\le |\sum_t \hat{f}_t(\hat{\theta}) - \sum_t f_{\mathcal{A}_t}(\hat{\theta})| + \max\left\{0, \sum_t \hat{f}_t(\hat{\theta})\right\},\end{aligned} \tag{EC.67}$$

where the first equality uses the fact that $f_{\mathcal{A}_t}(\cdot) \ge 0$.

Let $\Gamma_T := \max\left\{0, \sum_t \hat{f}_t(\hat{\theta})\right\}$ and $x = \sqrt{\sum_t f_{\mathcal{A}_t}(\hat{\theta})}$. We combine (EC.66) and (EC.67) to show that x^2, or equivalently $|\sum_t f_{\mathcal{A}_t}(\hat{\theta})|$, can be bounded as follows with probability $1-\mathcal{O}(T^{-1})$:

$$\begin{aligned} x^2 &\le \frac{2}{3}(C_3(s+m)+1)\log(T) + 3\sqrt{(C_3(s+m)+1)\log(T)}\cdot x + \Gamma_T \\ \Rightarrow x^2 - 3\sqrt{(C_3(s+m)+1)\log(T)}\cdot x &- (\Gamma_T + \frac{2}{3}(C_3(s+m)+1)\log(T)) \le 0.\end{aligned} \tag{EC.68}$$

Note that the inequality (EC.68) can be viewed as a quadratic function in x. Hence, we can solve for the upper bound of x:

$$\begin{aligned} x &\le \frac{3\sqrt{(C_3(s+m)+1)\log(T)} + \sqrt{9(C_3(s+m)+1)\log(T) + 4(\Gamma_T + \frac{2}{3}(C_3(s+m)+1)\log(T))}}{2} \\ &= \frac{3\sqrt{(C_3(s+m)+1)\log(T)} + \sqrt{(9+8/3)(C_3(s+m)+1)\log(T) + 4\Gamma_T}}{2} \\ &< \frac{3\sqrt{(C_3(s+m)+1)\log(T)} + 4\sqrt{(C_3(s+m)+1)\log(T)} + 2\sqrt{\Gamma_T}}{2} \\ &\le \frac{7}{2}\sqrt{(C_3(s+m)+1)\log(T)} + \sqrt{\Gamma_T} \\ \Rightarrow \sqrt{\sum_t f_{\mathcal{A}_t}(\hat{\theta})} &\le 4\sqrt{(C_3(s+m)+1)\log(T)} + \sqrt{\Gamma_T} \\ \Rightarrow \sum_t f_{\mathcal{A}_t}(\hat{\theta}) &\le 32(C_3(s+m)+1)\log(T) + 2\Gamma_T,\end{aligned} \tag{EC.69, EC.70}$$

where in (EC.69) we first enlarge $(9+8/3)$ to 16 and then uses the fact that $\sqrt{a^2+b^2} \le a+b$ for $a,b \ge 0$, and in (EC.70) we uses the fact that $(a+b)^2 \le 2a^2+2b^2$ for all $a,b \in \mathbb{R}$. The remaining part follows by combing above results with Lemma EC.5.

EC.9.4. Lemma EC.4

LEMMA EC.4. *Denote the empirical version* $\hat{f}_t(\theta) = \log(p_{Q^T P_0^T \theta}(c_t)/p_{\beta^*}(c_t))$ *for* $t>0$. *If event* $\mathcal{E}_2(m,T)$ *holds and* $\max\{\|\theta - P_0 Q \beta^*\|, \mathcal{G}_1(m,T)\} \le \frac{\rho}{8Kx_{\max}}$, *then with probability* $1-\delta_4$ *we have*

$$\left|\sum_t [\hat{f}_t(\theta) - f_{\mathcal{A}_t}(\theta)]\right| \le \frac{2}{3}\log(1/\delta_4) + 3\sqrt{\log(1/\delta_4)\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta)}. \tag{EC.71}$$

Proof. We can construct a Doob's martingale $\{M(i), i=0,1,2,...,T\}$ as follow

$$M(i) = \mathbb{E}\left[\sum_t \hat{f}_t(\theta)|\mathcal{H}_i\right], i=1,2,...T \tag{EC.72}$$

Using Bernstein's inequality, we can show that for $\epsilon > 0$,

$$\begin{aligned} &\mathbb{P}\left(|M(T)-M(0)| \ge t\right) \le \exp\left(-\frac{t^2}{2k+2t/3}\right) \\ \Rightarrow &\mathbb{P}\left(\left|\sum_t [\hat{f}_t(\theta) - f_{\mathcal{A}_t}(\theta)]\right| \ge \epsilon\right) \le \exp\left(-\frac{\epsilon^2}{2k+2\epsilon/3}\right), \end{aligned} \tag{EC.73}$$

where $k \ge \sum_{i=1}^{T} Var[M(i)-M(i-1)|\mathcal{H}_{i-1}]$. Next, we will upper bound k.
First, we show that the mean different is zero:

$$\begin{aligned} &\mathbb{E}[M(i)-M(i-1)|\mathcal{H}_{i-1}] \\ &= \mathbb{E}\left[\mathbb{E}[\sum_t \hat{f}_t(\theta)|\mathcal{H}_i] - \mathbb{E}[\sum_t \hat{f}_t(\theta)|\mathcal{H}_{i-1}]\right] \\ &= \mathbb{E}[\sum_t \hat{f}_t(\theta)] - \mathbb{E}[\sum_t \hat{f}_t(\theta)] = 0. \end{aligned}$$

As $\mathbb{E}[M(i)-M(i-1)|\mathcal{H}_{i-1}] = 0$, we can show that

$$\begin{aligned} &Var[M(i)-M(i-1)|\mathcal{H}_{i-1}] \\ &= \mathbb{E}[(M(i)-M(i-1)|\mathcal{H}_{i-1})^2] \\ &= \mathbb{E}[(\hat{f}_i(\theta) - f_{\mathcal{A}_i}(\theta))^2] \\ &= \mathbb{E}[\hat{f}_i(\theta)^2] - f_{\mathcal{A}_i}(\theta)^2, \end{aligned}$$

where the second-to-last equality follows from the fact that $\mathbb{E}[\hat{f}_i(\theta)] = f_{\mathcal{A}_t}(\theta)$. As we have $2x^2 > \log^2(1+x)$ for all $x > -1/2$. Then, when $(p_{Q^T P_0^T \theta}(j) - p_{\beta^*}(j))/(p_{\beta^*}(j)) \ge -1/2$ holds for all $j \in \mathcal{A}_t$, we have

$$\begin{aligned} \mathbb{E}[\hat{f}_t(\theta)^2] &= \sum_{j\in\mathcal{A}_t} p_{\beta^*}(j)\left(\log\left(\frac{p_{Q^T P_0^T \theta}(j)}{p_{\beta^*}(j)}\right)\right)^2 \\ &= \sum_{j\in\mathcal{A}_t} p_{\beta^*}(j)\left(\log\left(1+\frac{p_{Q^T P_0^T \theta}(j) - p_{\beta^*}(j)}{p_{\beta^*}(j)}\right)\right)^2 \end{aligned}$$

$$\leq 2\sum_{j\in\mathcal{A}_t}\frac{(p_{Q^TP_0^T\theta}(j)-p_{\beta^*}(j))^2}{p_{\beta^*}(j)}$$

In addition, we can also show that

$$\begin{aligned}
f_{\mathcal{A}_t}(\theta) &= \sum_{j\in\mathcal{A}_t} p_{\beta^*}(j)\log\left(\frac{p_{Q^TP_0^T\theta}(j)}{p_{\beta^*}(j)}\right)\\
&= \sum_{j\in\mathcal{A}_t} p_{\beta^*}(j)\left(\frac{p_{Q^TP_0^T\theta}(j)-p_{\beta^*}(j)}{p_{\beta^*}(j)} - \frac{1}{2(\xi_j)^2}\left(\frac{p_{Q^TP_0^T\theta}(j)-p_{\beta^*}(j)}{p_{\beta^*}(j)}\right)^2\right) \qquad \text{(EC.74)}\\
&= \sum_{j\in\mathcal{A}_t}(p_{Q^TP_0^T\theta}(j)-p_{\beta^*}(j)) + \sum_j \frac{1}{2(\xi_j)^2}\frac{(p_{Q^TP_0^T\theta}(j)-p_{\beta^*}(j))^2}{p_{\beta^*}(j)} \qquad \text{(EC.75)}\\
&= \sum_{j\in\mathcal{A}_t}\frac{1}{2(\xi_j)^2}\frac{(p_{Q^TP_0^T\theta}(j)-p_{\beta^*}(j))^2}{p_{\beta^*}(j)^2}\\
&\geq \frac{1}{2\max_j(\xi_j)^2}\sum_{j\in\mathcal{A}_t} p_{\beta^*}(j)\frac{(p_{Q^TP_0^T\theta}(j)-p_{\beta^*}(j))^2}{p_{\beta^*}(j)^2}\\
&\geq \frac{1}{2\max_j(\xi_j)^2}\mathbb{E}[\hat{f}_t(\theta)^2], \qquad \text{(EC.76)}
\end{aligned}$$

where (EC.74) uses the Taylor's expansion of $\log(a)$ at $a=1$ and (EC.75) uses $\sum_j p_{\beta^*}(j)=\sum_j p_{Q^TP_0^T\theta}(j)=1$. We then analyze the value of ξ_j. Since we expend the $\log(a)$ function at $a=1$, there exists an α_j such that

$$\begin{aligned}
\xi_j &= \alpha_j + (1-\alpha_j)\frac{p_{Q^TP_0^T\theta}(j)}{p_{\beta^*}(j)}\\
&= 1+(1-\alpha)\frac{p_{Q^TP_0^T\theta}(j)-p_{\beta^*}(j)}{p_{\beta^*}(j)}.
\end{aligned}$$

Claim: If $\max\{\|\theta-P_0Q\beta^*\|,\mathcal{G}_1(m,T)\}\leq\frac{\rho}{8Kx_{\max}}$, then $-\frac{1}{2}\leq\frac{p_{Q^TP_0^T\theta}(j)-p_{\beta^*}(j)}{p_{\beta^*}(j)}\leq\frac{1}{2}$ for all j.

$$\begin{aligned}
|p_{\beta^*}(j)-p_{Q^TP_0^T\theta}(j)| &= |p_{\beta^*}(j)-p_{\Sigma\beta^*}(j)+p_{\Sigma\beta^*}(j)-p_{Q^TP_0^T\theta}(j)|\\
&\leq |p_{\Sigma\beta^*}(j)-p_{Q^TP_0^T\theta}(j)| + |p_{\beta^*}(j)-p_{\Sigma\beta^*}(j)|\\
&\leq \|\nabla p_{\nu_1}(j)\|\|Q^TP_0\theta-\Sigma\beta^*\| + \|\nabla p_{\nu_2}(j)\|\|\beta^*-\Sigma\beta^*\|, \qquad \text{(EC.77)}
\end{aligned}$$

where ν_1 and ν_2 are in between $\{Q^TP_0\theta,\Sigma\beta^*\}$ and $\{\beta^*,\Sigma\beta^*\}$ respectively. We now analyze the upper bound of $\|\nabla p_\xi(j)\|$ as follow

$$\begin{aligned}
\nabla p_\nu(j) &= \frac{\exp(x_j^T\nu)x_j}{\sum\exp(x^T\nu)} - \frac{\exp(x_j^T\nu)\sum\exp(x^T\nu)x}{(\sum\exp(x^T\nu))^2}\\
&= p_\nu(j)\sum_{i\in\mathcal{A}_t}(x_j-p_\nu(i)x_i)\\
\Rightarrow \|\nabla p_\nu(j)\| &\leq 2Kx_{\max}, \qquad \text{(EC.78)}
\end{aligned}$$

Combining (EC.77), (EC.78), event $\mathcal{E}_2(m,T)$ and $\max\{\|\theta-P_0Q\beta^*\|,\mathcal{G}_1(m,T)\}\leq\frac{\rho}{8Kx_{\max}}$, then we have

$$\begin{aligned}
|p_{\beta^*}(j)-p_{Q^TP_0^T\theta}(j)| &= |\leq 2Kx_{\max}\|\theta-P_0Q\beta^*\| + 2Kx_{\max}\|(\Sigma-I)\beta^*\|\\
&\leq 2Kx_{\max}\cdot\frac{\rho}{8Kx_{\max}} + 2Kx_{\max}\mathcal{G}_1(m,T)
\end{aligned}$$

$$
\begin{aligned}
&\leq \frac{\rho}{4} + 2Kx_{\max}\frac{\rho}{8Kx_{\max}} \\
&\leq \frac{\rho}{2} \leq \frac{1}{2}p_{\beta^*}(j) \\
\Rightarrow &\frac{|p_{Q^T P_0^T \theta}(j) - p_{\beta^*}(j)|}{p_{\beta^*}(j)} \leq \frac{1}{2} \\
\Rightarrow &-\frac{1}{2} \leq \frac{p_{Q^T P_0^T \theta}(j) - p_{\beta^*}(j)}{p_{\beta^*}(j)} \leq \frac{1}{2},
\end{aligned}
$$

which proves the Claim. Hence, if we have $(p_{Q^T P_0^T \theta}(j) - p_{\beta^*}(j))/p_{\beta^*}(j) \leq 1/2$, then we can show that

$$
\xi_j \leq 1 + \frac{1}{2} = \frac{3}{2} \Rightarrow 2\max_j(\xi)^2 \leq \frac{9}{2}.
$$

Therefore, if we set k as follows:

$$
k = \sum_{i=1}^{T} \frac{9}{2} f_{\mathcal{A}_t}(\theta) \geq \sum_{i=1}^{T} \mathbb{E}[\hat{f}_i(\theta)] \geq \sum_{i=1}^{T} Var[M(i) - M(i-1)|\mathcal{H}_{i-1}]
$$

then from EC.73, we have

$$
\mathbb{P}\left(\left| \sum_t [\hat{f}_t(\theta) - f_{\mathcal{A}_t}(\theta)] \right| \geq \epsilon \right) \leq \exp\left(-\frac{\epsilon^2}{9\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta) + 2\epsilon/3} \right). \tag{EC.79}
$$

Finally, to ensure $\mathbb{P}\left(\left| \sum_t [\hat{f}_t(\theta) - f_{\mathcal{A}_t}(\theta)] \right| \geq \epsilon \right) \leq \delta$, we can set $\delta_4 = \exp\left(-\frac{\epsilon^2}{9\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta) + 2\epsilon/3} \right)$. We then solve for the ϵ.

$$
\begin{aligned}
\delta_4 &= \exp\left(-\frac{\epsilon^2}{9\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta) + 2\epsilon/3} \right) \\
\log(1/\delta_4) &= \frac{\epsilon^2}{9\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta) + 2\epsilon/3} \\
\log(1/\delta_4)(9\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta) + 2\epsilon/3) &= \epsilon^2 \\
\epsilon^2 - \frac{2}{3}\log(1/\delta_4)\epsilon - 9\log(1/\delta_4)\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta) &= 0 \\
\Rightarrow \frac{\frac{2}{3}\log(1/\delta_4) + \sqrt{(\frac{2}{3}\log(1/\delta_4))^2 + 36\log(1/\delta_4)\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta)}}{2} &= \epsilon \\
\Rightarrow \frac{\frac{2}{3}\log(1/\delta_4) + \frac{2}{3}\log(1/\delta_4) + 6\sqrt{\log(1/\delta_4)\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta)}}{2} &\geq \epsilon \\
\Rightarrow \frac{2}{3}\log(1/\delta_4) + 3\sqrt{\log(1/\delta_4)\sum_{i=1}^{T} f_{\mathcal{A}_t}(\theta)} &\geq \epsilon.
\end{aligned}
$$

The Lemma follows directly by plugging the last inequality back to EC.79.

EC.9.5. Lemma EC.5

LEMMA EC.5. *Let $\delta = \|\theta - P_0 Q\beta^*\|$ and C_3 be a positive constant. If events $\mathcal{E}_2(m,T)$ and $\mathcal{E}_{rp}(m,d,1/2)$ hold and $\delta \leq \frac{3}{4}\frac{n_T \mu}{T L_3}$, then the following inequality holds*

$$
\sum_t f_{\mathcal{A}_t}(\theta) \geq \frac{1}{4}(\theta - P_0 Q\beta^*)^T \left(\sum_t \nabla^2 f_{\mathcal{A}_t}(P_0 Q\beta^*) \right)(\theta - P_0 Q\beta^*)
$$

$$- 4x_{\max}^2 T\mathcal{G}_1(m,T)\delta. \tag{EC.80}$$

Proof. We first expend $\sum_t f_{\mathcal{A}_t}(\theta)$ at $P_0Q\beta^*$.

$$\sum_t f_{\mathcal{A}_t}(\theta) \geq \overbrace{\sum_t f_{\mathcal{A}_t}(P_0Q\beta^*)}^{\text{ⓐ}} + \overbrace{\sum_t \nabla f_{\mathcal{A}_t}(P_0Q\beta^*)^T(\theta - P_0Q\beta^*)}^{\text{ⓑ}}$$
$$+ \overbrace{\frac{1}{2}(\theta - P_0Q\beta^*)^T\left(\sum_t \nabla^2 f_{\mathcal{A}_t}(P_0Q\beta^*)\right)(\theta - P_0Q\beta^*) - \frac{1}{6}\sum_t L_3\|\theta - P_0Q\beta^*\|^3}^{\text{ⓒ}}. \tag{EC.81}$$

We will derive the lower bounds for these three parts in equation (EC.81).

Lower bound for ⓐ. Since $\sum_t f_{\mathcal{A}_t}(P_0Q\beta^*)$ can be viewed as KL-Divergence, we have

$$f_{\mathcal{A}_t}(P_0Q\beta^*) \geq 0. \tag{EC.82}$$

Lower bound for ⓑ. Via Cauchy inequality we have:

$$\nabla f_{\mathcal{A}_t}(P_0Q\beta^*)^T(\theta - P_0Q\beta^*) \geq -\|\nabla f_{\mathcal{A}_t}(P_0Q\beta^*)\|\|\theta - P_0Q\beta^*\| = -\|\nabla f_{\mathcal{A}_t}(P_0Q\beta^*)\|\delta. \tag{EC.83}$$

The remaining task is to find the bound for $\|\nabla f_{\mathcal{A}_t}(P_0Q\beta^*)\|$.

Claim: $\|\nabla f_{\mathcal{A}_t}(P_0Q\beta^*)\| \leq 6K^2x_{\max}^2\mathcal{G}_1(m,t)$ holds for all t.

$$\begin{aligned}
\nabla f_{\mathcal{A}_t}(P_0Q\beta^*) &= \sum_{j\in\mathcal{A}_t} \frac{p_{\beta^*(j)}}{p_{\Sigma\beta^*}(j)}\nabla p_{\Sigma\beta^*}(j) \\
&= \sum_{j\in\mathcal{A}_t} \frac{p_{\beta^*(j)}}{p_{\Sigma\beta^*}(j)} \frac{(\sum\exp(x^T\Sigma\beta^*))\exp(x_j^T\Sigma\beta^*)P_0Qx_j}{(\sum\exp(x^T\Sigma\beta^*))^2} \\
&- \sum_{j\in\mathcal{A}_t} \frac{p_{\beta^*(j)}}{p_{\Sigma\beta^*}(j)} \frac{\exp(x_j^T\Sigma\beta^*)\sum\exp(x^T\Sigma\beta^*)P_0Qx}{(\sum\exp(x^T\Sigma\beta^*))^2} \\
&= \sum_{j\in\mathcal{A}_t} \frac{p_{\beta^*(j)}}{p_{\Sigma\beta^*}(j)} p_{\Sigma\beta^*}(j)P_0Qx_j \\
&- \sum_{j\in\mathcal{A}_t} \frac{p_{\beta^*(j)}}{p_{\Sigma\beta^*}(j)} \frac{p_{\Sigma\beta^*}(j)\sum\exp(x^T\Sigma\beta^*)P_0Qx}{\sum\exp(x^T\Sigma\beta^*)} \\
&= \sum_{j\in\mathcal{A}_t} p_{\beta^*(j)}P_0Qx_j - \sum_{j\in\mathcal{A}_t} p_{\beta^*(j)} \sum_{j\in\mathcal{A}_t} p_{\Sigma\beta^*}(j)P_0Qx_j \\
&= \sum_{j\in\mathcal{A}_t} p_{\beta^*(j)}\left(P_0Qx_j - \sum_{i\in\mathcal{A}_t} p_{\Sigma\beta^*}(i)P_0Qx_i\right) \\
&= \sum_{j\in\mathcal{A}_t} (p_{\beta^*}(j) - p_{\Sigma\beta^*}(j))P_0Qx_j \\
&= \sum_{j\in\mathcal{A}_t} \nabla p_\xi(j)^T(\beta^* - \Sigma\beta^*)P_0Qx_j,
\end{aligned} \tag{EC.84}$$

where ξ is on the line between β^* and $\Sigma\beta^*$.

We can show that

$$\begin{aligned}
\nabla p_\xi(j) &= \frac{\exp(x_j^T\xi)x_j}{\sum\exp(x^T\xi)} - \frac{\exp(x_j^T\xi)\sum\exp(x^T\xi)x}{(\sum\exp(x^T\xi))^2} \\
&= p_\xi(j)\sum_{i\in\mathcal{A}_t}(x_j - p_\xi(i)x_i)
\end{aligned}$$

$$\Rightarrow \|\nabla p_\xi(j)\| \le 2Kx_{\max},$$

where we use the fact that $0 \le p_\xi(j) \le 1$ for all $j \in \mathcal{A}_t$. Via event $\mathcal{E}_2(m,T)$, we have

$$\begin{aligned}\|\nabla f_{\mathcal{A}_t}(P_0Q\beta^*)\| = \|\sum_{j\in\mathcal{A}_t} \nabla p_\xi(j)\,(\beta^* - \Sigma\beta^*)\,P_0Qx_j\| &\le K\cdot 2Kx_{\max}\cdot\|\beta^*-\Sigma\beta^*\|\max_{j\in\mathcal{A}_t}\|P_0Qx_j\| \\ &= 2K^2x_{\max}\mathcal{G}_1(m,T)\max_{j\in\mathcal{A}_t}\|P_0Qx_j\|.\end{aligned}$$

To prove the Claim, we finally need to bound $\|P_0Qx_j\|$ with $\|x_j\|$. Let $\tilde{x}_j = Qx_j$.

$$\begin{aligned}\|P_0Qx_j\| &= \|P_0\tilde{x}_j\| \\ &= \left\|\begin{pmatrix} I & \\ & P\end{pmatrix}\begin{pmatrix}\tilde{x}_{j1}\\ \tilde{x}_{j2}\end{pmatrix}\right\| \\ &= \left\|\begin{pmatrix}\tilde{x}_{j1}\\ P\tilde{x}_{j2}\end{pmatrix}\right\| \le \|\tilde{x}_{j1}\| + \|P\tilde{x}_{j2}\|.\end{aligned}$$

As event $\mathcal{E}_{rp}(m,d,1/2)$ holds, we have $\|P\tilde{x}_{j2}\| \le (1+1/2)\|\tilde{x}_{j2}\| \le 2\|\tilde{x}_{j2}\|$. Combining this result with the fact that Q is a permutation matrix that won't change the scale of x_j, we have

$$\|P_0Qx_j\| \le 3x_{\max}$$

holds with high probability. Therefore, we have

$$\|\nabla f_{\mathcal{A}_t}(P_0Q\beta^*)\| \le 2Kx_{\max}\mathcal{G}_1(m,t)\max_{j\in\mathcal{A}_t}\|P_0Qx_j\| \le 6K^2x_{\max}^2\mathcal{G}_1(m,t),$$

which proves the Claim.

Thus, applying this Claim for all t, we have

$$\begin{aligned}&\|\sum_t \nabla f_{\mathcal{A}_t}(P_0Q\beta^*)\| \le 6TK^2x_{\max}^2\mathcal{G}(T) \\ \Rightarrow &\sum_t \nabla f_{\mathcal{A}_t}(P_0Q\beta^*)^T(\theta - P_0Q\beta^*) \ge -6TK^2x_{\max}^2\mathcal{G}(T)\delta. \end{aligned} \tag{EC.85}$$

Bound for © By Lemma EC.6 and Assumption A.3, with probability $1-\mathcal{O}(1/T)$ we have

$$\begin{aligned}&\frac{1}{2}(\theta - P_0Q\beta^*)^T\left(\sum_t \nabla^2 f_{\mathcal{A}_t}(P_0Q\beta^*)\right)(\theta - P_0Q\beta^*) \\ &\ge \frac{1}{2}(\theta - P_0Q\beta^*)^T\left(\sum_{t\in\mathcal{W}_R} \nabla^2 f_{\mathcal{A}_t}(P_0Q\beta^*)\right)(\theta - P_0Q\beta^*) \\ &\ge \frac{1}{4}\mu n_T\|\theta - P_0Q\beta^*\|^2. \end{aligned} \tag{EC.86}$$

Since we require $\delta \le \frac{3}{4}\frac{n_T\mu}{TL_3}$, we can further show that

$$\begin{aligned}&\frac{1}{6}L_3\|\theta - P_0Q\beta^*\|^3 \le \frac{n_T\mu}{8T}\|\theta - P_0Q\beta^*\|^2 \\ &\le \frac{1}{4T}(\theta - P_0Q\beta^*)^T\left(\sum_t \nabla^2 f_{\mathcal{A}_t}(P_0Q\beta^*)\right)(\theta - P_0Q\beta^*) \\ \Rightarrow &\frac{1}{2}(\theta - P_0Q\beta^*)^T\left(\sum_t \nabla^2 f_{\mathcal{A}_t}(P_0Q\beta^*)\right)(\theta - P_0Q\beta^*) - \frac{1}{6}\sum_t L_3\|\theta - P_0Q\beta^*\|^3\end{aligned}$$

$$\geq \frac{1}{4}(\theta - P_0 Q\beta^*)^T \left(\sum_t \nabla^2 f_{\mathcal{A}_t}(P_0 Q\beta^*)\right)(\theta - P_0 Q\beta^*). \tag{EC.87}$$

EC.9.6. Lemma EC.6

LEMMA EC.6. *Under Assumption A.3, for all feasible θ in the projected space, if $n_T = \mathcal{O}(2(s+m)^2(\log T - \log(s+m)))$, then with probability at least $1-\mathcal{O}(1/T)$, we have*

$$\sum_{t=1}^{T} \nabla^2 f(\theta) \succeq \frac{1}{2}\mu n_T I.$$

Proof. Since $\nabla^2 f(\hat{\theta})$ is always positive semidefinite, we will have

$$\sum_{t=1}^{T} \nabla^2 f(\hat{\theta}) \succeq \sum_{i\in\mathcal{W}_R} \nabla^2 f(\hat{\theta}).$$

From Lemma EC.8, we have

$$\nabla^2 f(\hat{\theta}) = \tilde{\mathbb{E}}\left[\left(z - \tilde{\mathbb{E}}[z]\right)\left(z - \tilde{\mathbb{E}}[z]\right)^T\right] \preceq \tilde{\mathbb{E}}[zz^T] \preceq (s+m) z_{\max}^2 I. \tag{EC.88}$$

Combining this inequality with the fact that 1) $i \in \mathcal{W}_R$ are i.i.d random sample and 2) $\nabla^2 f(\hat{\theta})$ is always positive semidefinite, we can use the Matrix Chernoff inequalities to show that

$$\mathbb{P}\left(\lambda_{\min}\left(\sum_{i\in\mathcal{W}_R}^{T} \nabla^2 f(\hat{\theta})\right) \leq (1-\delta)\mu_{\min}\right) \leq (s+m)\left(\frac{e^{-\delta}}{(1-\delta)^{1-\delta}}\right)^{\mu_{\min}/R}, \tag{EC.89}$$

where $\mu_{\min} \leq \lambda_{\min}\left(\sum_{i\in\mathcal{W}_R}^{T} \mathbb{E}[\nabla^2 f(\hat{\theta})]\right)$ and $R \geq \lambda_{\max}(\nabla^2 f(\hat{\theta}))$. Under assumption A.3, we can show that $\lambda_{\min}\left(\sum_{i\in\mathcal{W}_R}^{T} \mathbb{E}[\nabla^2 f(\hat{\theta})]\right) \geq n_T\mu$. Based on (EC.88), we can show that $\nabla^2 f(\hat{\theta}) \preceq (s+m) z_{\max}^2 I$. Hence, we set $\mu_{\min} = n_T\mu$ and $R = (s+m)z_{\max}^2$.

Moreover, if we pick $\delta = 1/2$, we then have

$$\begin{aligned}
\mathbb{P}\left(\lambda_{\min}\left(\sum_{i\in\mathcal{W}_R}^{T} \nabla^2 f(\hat{\theta})\right) \leq \frac{1}{2}n_T\mu\right) &\leq (s+m)\left(\frac{e^{-1/2}}{(1/2)^{1/2}}\right)^{n_T\mu/(s+m)z_{\max}^2} \\
&= (s+m)\left(\frac{e}{2}\right)^{-n_T\mu/2(s+m)z_{\max}^2} \\
&= (s+m)\left(\frac{e}{2}\right)^{-\frac{n_T\mu}{2(s+m)z_{\max}^2}}.
\end{aligned}$$

The remaining part of this lemma follows directly by using $n_T \gtrsim \frac{2(s+m)z_{\max}^2(\log T - \log(s+m))}{\mu\log(e/2)}$.

EC.9.7. Lemma EC.7

LEMMA EC.7. *Let $B_t = (\sum_i^t \nabla^2 f_{\mathcal{A}_i}(P_0 Q\beta^*))^{-1/2} \nabla^2 f_{\mathcal{A}_t}(P_0 Q\beta^*)(\sum_t \nabla^2 f(P_0 Q\beta^*))^{-1/2}$. Under Assumptions A.1, A.2, and A.3, with probability $1-\mathcal{O}(1/T)$, we have*

$$\sum_{t=T_0+1}^{T} \min\{1, \|B_t\|_{op}\} \leq 2(s+m)\log\left(\frac{8TK^2x_{\max}^2}{\mu n_{T_0}}\right). \tag{EC.90}$$

Proof. Denote the eigenvalue as $\sigma_1(B_t) \geq \sigma_2(B_T) \geq ... \geq 0$ and we can show that

$$\sum_{t=T_0+1}^{T} \nabla^2 f(P_0 Q\beta^*) = \sum_{t=T_0+1}^{T-1} \nabla^2 f(P_0 Q\beta^*) + \nabla^2 f_{\mathcal{A}_t}(P_0 Q\beta^*)$$

$$= (\sum_{t=T_0+1}^{T-1} \nabla^2 f(P_0 Q\beta^*))^{1/2}(I+B_T)(\sum_{t=T_0+1}^{T-1} \nabla^2 f(P_0 Q\beta^*))^{1/2}$$

$$\Rightarrow \log\left(\frac{\det\sum_{t=T_0+1}^{T}\nabla^2 f(P_0Q\beta^*)}{\det\sum_{t=T_0+1}^{T-1}\nabla^2 f(P_0Q\beta^*)}\right) = \sum_j \log(1+\sigma_j(B_T)) \geq \log(1+\sigma_1(B_T)),$$

Together with the observation that $2\log(1+x) \geq x$ for $x \in (0,1]$, we can show the following inequality holds:

$$\begin{aligned}\min\{1, \|B_T\|_{op}\} &\leq 2\log(1+\|B_t\|_{op}) \\ &= 2\log(1+\sigma_1(B_t)) \\ &\leq 2\log\left(\frac{\det\sum_{t=T_0+1}^{T}\nabla^2 f(P_0Q\beta^*)}{\det\sum_{t=T_0+1}^{T-1}\nabla^2 f(P_0Q\beta^*)}\right) \end{aligned} \tag{EC.91}$$

Therefore, we can prove that

$$\sum_{t=T_0+1}^{T} \min\{1, \|B_t\|_{op}\} \leq 2\log\left(\frac{\det\sum_{t=1}^{T}\nabla^2 f(P_0Q\beta^*)}{\det\sum_{t=1}^{T_0+1}\nabla^2 f(P_0Q\beta^*)}\right).$$

From Lemma EC.6, we can show with probability $1-\mathcal{O}(1/T)$, the following inequality holds:

$$\sum_t^T \nabla^2 f(P_0Q\beta^*) \succeq \frac{1}{2}\mu n_T.$$

Then, following the similar procedures as in (EC.57) in Lemma EC.1, we can show that

$$\sum_t^T \nabla^2 f(P_0Q\beta^*) \preceq 4TK^2x_{\max}^2.$$

Hence,

$$\begin{aligned} &2\log\left(\frac{\det\sum_{t=1}^{T}\nabla^2 f(P_0Q\beta^*)}{\det\sum_{t=1}^{T_0+1}\nabla^2 f(P_0Q\beta^*)}\right) \leq 2(s+m)\log\left(\frac{8TK^2x_{\max}^2}{\mu n_{T_0}}\right) \\ \Rightarrow &\sum_{t=T_0+1}^{T} \min\{1, \|B_t\|_{op}\} \leq 2(s+m)\log\left(\frac{8TK^2x_{\max}^2}{\mu n_{T_0}}\right). \end{aligned}$$

EC.9.8. Lemma EC.8

LEMMA EC.8. $\nabla^2 f_{\mathcal{A}}(\hat{\theta}) = \tilde{\mathbb{E}}\left[\left(z-\tilde{\mathbb{E}}[z]\right)\left(z-\tilde{\mathbb{E}}[z]\right)^T\right]$.

Proof. We first consider the gradient of $f_{\mathcal{A}}(\theta)$:

$$\nabla f_{\mathcal{A}}(\theta) = \sum_{j\in\mathcal{A}} p_{\Sigma\beta^*,\mathcal{A}}(j)\frac{\nabla p_{Q^TP_0^T\theta,\mathcal{A}}(j)}{p_{Q^TP_0^T\theta,\mathcal{A}}(j)}. \tag{EC.92}$$

Then, we compute the term $\nabla p_{Q^TP_0^T\theta,\mathcal{A}}(j)$:

$$\begin{aligned}\nabla p_{Q^TP_0^T\theta,\mathcal{A}}(j) &= \frac{(\sum_{i\in\mathcal{A}}\exp(z_i^T\theta)\exp(z_j^T\theta)z_j - \exp(z_j^T\theta)\sum_{i\in\mathcal{A}}\exp(z_i^T\theta)z_i}{(\sum_{i\in\mathcal{A}}\exp(z_i^T\theta))^2} \\ &= \frac{\exp(z_j^T\theta)z_j}{\sum_{i\in\mathcal{A}}\exp(z_i^T\theta)} - \frac{\exp(z_j^T\theta)}{\sum_{i\in\mathcal{A}}\exp(z_i^T\theta)}\cdot\sum_{i\in\mathcal{A}}\frac{\exp(z_i^T\theta)z_i}{\sum_{i\in\mathcal{A}}\exp(z_i^T\theta)} \\ &= p_{Q^TP_0^T\theta,\mathcal{A}}(j)z_j - p_{Q^TP_0^T\theta,\mathcal{A}}(j)\sum_{i\in\mathcal{A}} p_{Q^TP_0^T\theta,\mathcal{A}}(i)z_i, \end{aligned}$$

which implies that $\frac{\nabla p_{Q^T P_0^T \theta,\mathcal{A}}(j)}{p_{Q^T P_0^T \theta,\mathcal{A}}(j)} = z_j - \sum_{i\in\mathcal{A}} p_{Q^T P_0^T \theta,\mathcal{A}} z_i$. Combining it with (EC.92), we will have

$$\begin{aligned}\nabla f_{\mathcal{A}}(\theta) &= \sum_{j\in\mathcal{A}} p_{\Sigma\beta^*,\mathcal{A}}(j)\left(z_j - \sum_{i\in\mathcal{A}} p_{Q^T P_0^T \theta,\mathcal{A}} z_i\right)\\ &= \sum_{j\in\mathcal{A}} p_{\Sigma\beta^*,\mathcal{A}}(j) z_j - \sum_{j\in\mathcal{A}} p_{\Sigma\beta^*,\mathcal{A}}(j)\sum_{i\in\mathcal{A}} p_{Q^T P_0^T \theta,\mathcal{A}} z_i\\ &= \sum_{j\in\mathcal{A}} p_{\Sigma\beta^*,\mathcal{A}}(j) z_j - \sum_{i\in\mathcal{A}} p_{Q^T P_0^T \theta,\mathcal{A}} z_i.\end{aligned}$$

Therefore, we can show that

$$\begin{aligned}\nabla^2 f_{\mathcal{A}}(\theta) &= \sum_{i\in\mathcal{A}} \nabla p_{Q^T P_0^T \theta,\mathcal{A}}(i) z_i^T\\ &= \sum_{i\in\mathcal{A}} p_{Q^T P_0^T \theta,\mathcal{A}}(i) z_i z_i^T - \sum_{i\in\mathcal{A}} p_{Q^T P_0^T \theta,\mathcal{A}}(i)\sum_{k\in\mathcal{A}} p_{Q^T P_0^T \theta,\mathcal{A}}(k) z_k z_i^T\\ &= \sum_{i\in\mathcal{A}} p_{Q^T P_0^T \theta,\mathcal{A}}(i) z_i z_i^T - \sum_{k\in\mathcal{A}} p_{Q^T P_0^T \theta,\mathcal{A}}(k) z_k \sum_{i\in\mathcal{A}} p_{Q^T P_0^T \theta,\mathcal{A}}(i) z_i^T\\ &= \tilde{\mathbb{E}}[zz^T] - \tilde{\mathbb{E}}[z]\tilde{\mathbb{E}}[z^T]\\ &= \tilde{\mathbb{E}}\left[zz^T - \tilde{\mathbb{E}}[z]\tilde{\mathbb{E}}[z^T] + z\tilde{\mathbb{E}}[z^T] - \tilde{\mathbb{E}}[z]z^T\right]\\ &= \tilde{\mathbb{E}}\left[\left(z - \tilde{\mathbb{E}}[z]\right)\left(z - \tilde{\mathbb{E}}[z]\right)^T\right],\end{aligned}$$

where we use the definition of $\tilde{\mathbb{E}}(\cdot)$ in the last three equations.

EC.9.9. Lemma EC.9

LEMMA EC.9. *If we adopt the random sampling schedule with the success probability* $P_{C_0}(t) = \min\left\{1, C_0\left[\frac{1}{t^{1/2}} + \frac{\mathbb{1}(\mathcal{G}_0 \le 3\beta_{\min})}{t^{1/3}}\right]\right\}$, *where* $C_0 > 0$, *the following two statement holds*

1. *When* $\mathcal{G}_0(T) \ge \frac{1}{3}\beta_{\min}$, *with probability* $1 - 2\exp(-\frac{C_0}{10}(T-1)^{2/3})$, $n_T = \mathcal{O}(C_0 T^{2/3})$.
2. *When* $\mathcal{G}_0(T) < \frac{1}{3}\beta_{\min}$, *with probability* $1 - 2\exp(-\frac{C_0}{10}(T-1)^{1/2})$, $n_T = \mathcal{O}(C_0 T^{1/2})$.

Proof. At time T, the expected total number of random decisions is

$$\mathbb{E}[n_T] = \sum_{t=1}^{T} \min\left\{1, C_0\left[\frac{1}{t^{1/2}} + \frac{\mathbb{1}(\mathcal{G}_0(T) \le 3\beta_{\min})}{t^{1/3}}\right]\right\}.$$

When $T \ge \lfloor 8C_0^3 + 1\rfloor := T_0$,

$$\mathbb{E}[n_T] = T_0 + C_0 \sum_{t=T_0}^{T}\left[\frac{1}{t^{1/2}} + \frac{\mathbb{1}(\mathcal{G}_0(T) \le 3\beta_{\min})}{t^{1/3}}\right].$$

We can verify that for any $\alpha \in (0,1)$

$$(T-1)^{1-\alpha} - (T_0-1)^{1-\alpha} \le \sum_{t=T_0}^{T} t^{-\alpha} \le T^{1-\alpha} - T_0^{1-\alpha}.$$

Hence, when $\mathcal{G}_0(T) \ge \frac{1}{3}\beta_{\min}$:

$$(T-1)^{2/3} - (T_0-1)^{2/3} \le \frac{\mathbb{E}[n_T] - T_0}{C_0} \le 2(T^{2/3} - T_0^{2/3}). \tag{EC.93}$$

When $\mathcal{G}_0(T) < \frac{1}{3}\beta_{\min}$:

$$(T-1)^{1/2} - (T_0-1)^{1/2} + T_{poor} \leq \frac{\mathbb{E}[n_T] - T_0}{C_0} \leq T^{1/2} - T_0^{1/2} + T_{poor}, \tag{EC.94}$$

where $T_{poor} = \sum_{t=T_0}^{T} \mathbb{1}(\mathcal{G}(t) \leq 3\beta_{\min}) t^{-1/3}$. Further, using Chernoff bound, we have

$$\mathbb{P}\left(\frac{1}{2}\mathbb{E}[n_T] \leq n_T \leq \frac{3}{2}\mathbb{E}[n_T]\right) \geq 1 - 2\exp\left(-\frac{1}{10}\mathbb{E}[n_T]\right). \tag{EC.95}$$

Combining (EC.95) and (EC.93), we can conclude that when $\mathcal{G}_0(T) \geq \frac{1}{3}\beta_{\min}$:

$$\begin{aligned}
\mathbb{P}\left(\frac{1}{2}\left(T_0 + C_0((T-1)^{2/3} - (T_0-1)^{2/3})\right) \leq n_T \leq \frac{3}{2}\left(2C_0(T^{2/3} - T_0^{2/3}) + T_0\right)\right) \\
\geq 1 - 2\exp\left(-\frac{1}{10}\left(T_0 + C_0((T-1)^{2/3} - (T_0-1)^{2/3})\right)\right). \\
\Rightarrow \mathbb{P}(n_T = \mathcal{O}(T^{2/3})) \geq 1 - 2\exp(-\frac{C_0}{10}(T-1)^{2/3}).
\end{aligned} \tag{EC.96}$$

Similarly, when $\mathcal{G}_0(T) < \frac{1}{3}\beta_{\min}$, we have

$$\mathbb{P}(n_T = \mathcal{O}(T^{1/2})) \geq 1 - 2\exp(-\frac{C_0}{10}(T-1)^{1/2}). \tag{EC.97}$$