Survey and Paper Replication: Bayesian Meta-Prior Learning Using Empirical Bayes

Zihui Weng *
School of Mathematics
Sun Yat-sen University
Guangzhou,China
ryanwengzh@gmail.com

Abstract

1 Introduction

Recently, in tremendous scenarios one will encounter the situations where an agent must make successive decisions while interacting with an unknown environment in a sequential manner and obtains reward stimulus after taking action in each round. One could view this scenario as an optimization problem, whose objective is to maximize the expected cumulative reward over the entire sequence of decisions.

One application that encounters this scenario is the recommender system in e-commerce where agents will select a series of items in the widgets representing the contents of the page to maximize the expected reward from the consumers based on their multi-dimensional attributes such as browsing history including clicks, likes, dislikes, etc. Enterprises specializing in e-commerce show keen interest in developing algorithms to optimize the expected reward of the recommender system when facing the uncertainty of unknown consumers' preferences.

A hot issue in recommender system, which could be extracted into a problem in sequential decision-making scenarios, is the exploration-exploitation trade-off where the agent must balance the exploration of the unknown environment to mining new items that the consumers may show their preferences and exploitation of familiar fields to recommend optimal items to the consumers which match their preferences.

One could utilize the Multi-Armed Bandit method to cope with exploration-exploitation trade-off in recommender system. One of the classical approaches towards this dilemma is the Thompson Sampling method, which is a Bayesian approach that samples the next action and selects the optimal strategy according to the updated posterior distribution of the parameters. However, in real-world applications, original Thompson Sampling method is not applicable due to the misspecification of the prior distribution. Enormous literatures have showed the importance and the challenge of choosing an appropriate prior distribution, which reflects that we usually pick the non-informative prior distribution to avoid the bias of the prior distribution. Recent research further aims at searching for a good design of informative prior to seek better performance and decouple the learning rate of different categories instead of a consistent rate(i.e. same prior of different categories). The article by Nabi et.al provides a general solution in estimating empirical priors for models with Bayesian bandits or involving Bayesian learning using empirical bayes method. Empirical Bayes method makes statistical inference based on the observed data to estimate the prior distribution, which could be viewed as a pilot study towards the prior similar to the method of Poor Man Data Augmentation

*This report is accomplished in 2022 HKUST Summer Program of Department of Mathematics

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

based on the early randomized data to generate preliminary acquaintance of the data, manifesting good properties under specific assumptions.

The organization of this report is as follows: Section 2 introduces the methodology of the method. Section 3 reports the result of adult dataset simulations and MAB Live experiments. Section 4 concludes the report. Section 5 provides the extensions and individual method of this empirical bayes method.

Keywords: Bayesian Meta-Prior Learning, Empirical Bayes, Thompson Sampling, Multi-Armed Bandit, Recommender System

2 Methodology

In this section we will derive our method of obtaining an empirical bayes estimator under specific assumptions.

2.1 Assumptions and Derivations

The problem we focus on is the scenario where the features are grouped into disjoint sets in an arbitrary number such as the first-order features and second-order feature interactions in recommender system setting.

The assumptions are listed as follows in the non-overlapping feature grouping settings to propose a Bayesian hierarchical model. First we assume that the k^{th} category C_k has a distince hyperparameter meta-prior distribution which could be determined by experts' knowledge and here we assume them as Gaussian distribution, denoted by $N(\nu_k, \tau_k^2)$. In each category, the features' true effect μ_i (i.e. coefficients) are assumed to be independent and identically distributed from the corresponding category's meta-prior, i.e.

$$\mu_i \sim N(\nu_k, \tau_k^2) \quad \mathbb{E}[\mu_i] = \nu_k \quad \mathbb{V}[\mu_i] = \tau_k^2 \quad \forall i \in C_k \tag{1}$$

where notations $\mathbb{E}$ and $\mathbb{V}$ denote the expectation and variance of the random variable respectively.

Second, let $\tilde{\mu_i}$ and $\tilde{\sigma_i^2}$ is the estimators of $\mathbb{E}[\mu_i]$ and $\mathbb{V}[\mu_i]$ respectively and there exists a model to estimate them. We also assume that the first-order and second-order moments of $\tilde{\mu_i}$ conditional on the true effect μ_i are matched to the corresponding category's meta-prior and the expectation of $\tilde{\mu_i}$ equals to a pre-defined value ν_k, i.e.

$$\mathbb{E}\left[\tilde{\mu_i} \mid \mu_i\right] = \mu_i, \mathbb{V}\left[\tilde{\mu_i} \mid \mu_i\right] = \tilde{\sigma}_i^2, \mathbb{E}\left[\tilde{\mu_i}\right] = \nu_k, \quad \forall i \in C_k \tag{2}$$

Through equation (2), we obtain the variance of sample mean of the true effect for each feature in category C_k via variance decomposition:

$$\mathbb{V}\left[\tilde{\mu_i}\right] = \mathbb{E}\left[\mathbb{V}\left[\tilde{\mu_i} \mid \mu_i\right]\right] + \mathbb{V}\left[\mathbb{E}\left[\tilde{\mu_i} \mid \mu_i\right]\right] = \mathbb{E}\left[\tilde{\sigma}_i^2\right] + \tau_k^2, \quad \forall i \in C_k \tag{3}$$

Using equation (3) we could provide the formula of meta-prior variance τ_k^2:

$$\tau_k^2 = \mathbb{V}\left[\tilde{\mu_i}\right] - \mathbb{E}\left[\tilde{\sigma}_i^2\right] \tag{4}$$

Finally, we denote the response binary variable $y \in \{-1, 1\}$ and features as x. In the next section we will derive the meta-prior estimation through the assumptions above.

2.2 Derivation of Meta-Prior Hyperparameters Estimation

Now our target is to derive a sample-pathway estimator of meta-prior variance given equation (4). We further denote $\hat{\tau}_{k,t}^2$ as the meta-prior variance estimatior at time t and we have the following equation by applying the definition of sample mean and variance into equation (4):

$$\hat{\tau}_{k,t}^2 = \widehat{\mathbb{V}\left[\tilde{\mu}_{i,t}\right]} - \widehat{\mathbb{E}\left[\tilde{\sigma}_{i,t}^2\right]} = \frac{\sum_{i \in C_k} \left(\tilde{\mu}_{i,t} - \hat{\nu}_{k,t}\right)^2}{N_k - 1} - \frac{\sum_{i \in C_k} \tilde{\sigma}_{i,t}^2}{N_k} \tag{5}$$

where N_k denotes the number of features in category C_k and

$$\hat{\nu}_{k,t} = \frac{\sum_{i \in C_k} \tilde{\mu}_{i,t}}{N_k}, \quad \forall C_k \tag{6}$$

which could be interpreted as a combination of an unbiased estimator and an estimation noise $\frac{\sum_{i \in C_k} \tilde{\sigma}_{i,t}^2}{N_k}$.

Since the model we will propose in the next section requires the property of invariance to the feature sign changes, we set $\nu_k = 0$ to satisfy this property which cause little impact on the final result utilizing generalized linear model in arbitrary value setting of ν_k. Hence, we've obtain an additional degree of freedom and rewrite the equation (5) as:

$$\hat{\tau}_{k,t}^2 = \frac{\sum_{i \in C_k} \left[\tilde{\mu}_{i,t}^2 - \tilde{\sigma}_{i,t}^2\right]}{N_k}, \quad \forall C_k \tag{7}$$

Utilizing equation (7), the sample variance of the meta-prior is obtained and applied to the Bayesian Linear Probit Model in the next section as the variance of the resetted meta-prior.

2.3 Model Construction

The scenario we are interested in is the Bayesian generalized linear bandit, which uses an Gaussian distribution to the feature effects. The authors utilize the Bayesian Linear Probit model(BLIP)to learn the feature weights in a Bayesian manner. The model is defined as follows:

$$P(y|x, \tilde{\mu}) = \Phi(y \cdot \frac{\tilde{\mu}^T x}{\beta}) \tag{8}$$

where Φ denotes the cumulative distribution function of the standard normal distribution and β represents the steepness which we set to 1 in our experiments. This model assumes that the weights(i.e. coefficients of features) are mutually independent random variable and uses standard Gaussian prior $\mathcal{N}(0, 1)$ as the conjugate prior for the weights so that we could update the parameters of the weights' posterior distribution in component wise. This model guarantees the non-increased variance of each subsequent weight update and the weight variance is upper bounded by the initial pre-defined prior.

Combining the model with the meta-prior estimation in the previous section, the authors proposed the improved model as follows: we start the model with a non-informative prior $\mathcal{N}(0, 1)$ and train the model in batches. At an early time t, we compute the empirical bayes prior $\mathcal{N}(0, \tau_{k,t}^2)$ for each feature category C_k through equantion (7) and restart the model with the data-driven informative prior. Then we re-train the model with the original dataset to obtain the final estimation of the parameters.

Here we first provide necessary notations. We encode the sample in one-hot encoding way, i.e. $x := (x_1^T, ..., x_N^T) \quad x_i := (x_{i,1}, ..., x_{i,M_i}), \sum_{j=1}^{M_i} x_{i,j} = 1$. the means and variance of this vector are denoted as $\mu := (\mu_{1,1}, ..., \mu_{N,M_N})^T$ and $\sigma^2 := (\sigma_{1,1}^2, ..., \sigma_{N,M_N}^2)^T$ respectively. The update formula is as follows **??**:

$$\tilde{\mu}_{i,j} = \mu_{i,j} + y x_{i,j} \frac{\sigma_{i,j}^2}{\Sigma} v(\frac{y x^T \mu}{\Sigma}) \tag{9}$$

$$\tilde{\sigma}_{i,j}^2 = \sigma_{i,j}^2 (1 - x_{i,j} \frac{\sigma_{i,j}^2}{\Sigma^2} w(\frac{y x^T \mu}{\Sigma})) \tag{10}$$

where $\sum^2 := \beta^2 + x^T \sigma^2$ controls the learning rate of different categories and $v(t) := \frac{\mathcal{N}(t;0,1)}{\Phi(t;0,1)}$ and $w(t) := v(t)[v(t) + t]$.

It's worthnoting that according to the updating method proposed by **??** is a component-wise way, which is not suitable for the model we proposed with large-scale samples and the requirement of the fast online update for the sake of the low running speed of for loop in R. Hence, we turn to another update method proposed by **?**, which is a matrix-wise way and could be easily implemented in R.

Referring to **?**, the brief derivation of the matrix-wise update method is as follows. Let $Y_i^* = x_i^T\mu + \epsilon_i \sim^{i.i.d} \mathcal{N}(0,1)$ be a latent random variable corresponding to Y_i, which defines the structre of the estimation problem where Y_i equals to 1 when Y_i^* is non-negative and 0 otherwise. Augmenting this model with this latent variable, we could derive the likelihood contribution from observation i as:

$$p(y_i|y_i^*) = \mathbf{1}_{y_i=0}\mathbf{1}_{y_i<0} + \mathbf{1}_{y_i=1}\mathbf{1}_{y_i=\geq 0} \tag{11}$$

where $\mathbf{1}_A$ denotes the indicator function of interval A. We could further derive the posterior distribution and the conditional posterior distribution of the latent variable when taking the normal distribution as prior, i.e. $\mu \sim \mathcal{N}(\mu_0, B_0)$:

$$\pi(\mu, Y_i^*|y, X) \propto \sum_{i=1}^{n}[p(y_i|y_i^*)] \times N_N(Y^*|X\mu, I_N) \times N_K(\mu|\mu_0, B_0) \tag{12}$$

$$Y_i^*|\mu, y, X \sim TN_{[0,\infty)}(x_i^T\mu, 1), \quad y_i = 1 \tag{13}$$

$$Y_i^*|\mu, y, X \sim TN_{(-\infty,0)}(x_i^T\mu, 1), \quad y_i = 0 \tag{14}$$

where TN_A denotes the truncated normal density in interval A. Hence, the conditional posterior distribution of the weights(i.e. coefficients of features) is:

$$\mu|Y^*, X \sim N(\mu_n, B_n), \quad B_n = (B_0^{-1} + X^TX)^{-1}, \quad \mu_n = B_n(B_0^{-1}\mu_0 + X^TY^*) \tag{15}$$

Utilizing equation (13), we could update the weights in matrix-wise way and satisfy the requirement of the online update within a short time. The complete implementation of the matrix-wise update method is shown in Algorithm 2.

In the next section we will perform the experiments to evaluate the performance of the proposed model and compare it with the state-of-the-art methods i.e. BLIP model.

3 Simulations and Experiments

To validate the performance of BLIP model using meta-prior estimation, the authors apply it on two datasets, Adult dataset and MAB Live experiments.

3.1 Adult Dataset

We select the adult dataset from UCI Machine Learning Repository with the aim of predicting whether one's income will exceed 50K per year based on their background and attributes.

3.1.1 Data Preprocessing

We first omit the samples with missing values of features and obtain the initial train and test dataset with 30,162 and 15,060 samples respectively. Since the interactions of features are also importance factors that affect the prediction, we generate the second-order features through pairwise combinations of the component of the first-oder features and combine the first-order features and second-order features to form our raw feature vector. In fact, our strategy is to encode the categorical first features in one-hot encoding way and scale and center the numeric features to guarantee the vector is bounded in unit sphere under specific norm(e.g. L_2 norm) and then generate the second-order features by pairwise combinations of this one-hot encoded first-order features, with total 5K+ features.

Considering the high-dimension and sparsity of new dataset, it's necessary to perform feature selection to reduce the dimension of the dataset in order to pick out the most relevent features to get involved

in model training. Here we choose adaptive LASSO for the sake of its oracle properties of strong concsistency. The objective of adaptive LASSO is as follows:

$$min_W||y - X^T W||^2 + \lambda \sum_i \zeta_i |w_i| \tag{16}$$

where λ is the regularization parameter obtained through cross validation ,W represents the weights and ζ_i is the adaptive weight of w_i which is defined as: $\zeta_i = \frac{1}{|\tilde{w_i}|^\gamma}$ where $\tilde{w_i}$ is the weights estimators via ridge regression as follows:

$$min_W||y - X^T W||^2 + \lambda \sum_i |w_i| \tag{17}$$

After feature selection, there are two strategies to deal with the selected features. One is to collect all the second-order faeture components which contain the selected second-order features while another choice is to keep the results of feature selection. In order to obtain higher perfomance of the model, the latter one is selected.

We also split the train dataset into 6 batches, 5K instances for each batches and train the proposed model in batches. The test dataset is used to evaluate the performance of the model.

In the next step we propose three models and compare their performances.

3.1.2 Proposed Model

Initialized with a standard normal prior $\mathcal{N}(0, 1)$, the proposed models are as follows:

BLIP: We opt for Bayesian Linear Probit model as our baseline model and update the model in batches. The update method here is matrix-wise as mentioned in the previous section.

BLIPBayes: This is the proposed model with Bayesian meta-prior estimation. At a small time t, we train the BLIP model with the first t batches and obtain the posterior distribution of the weights. Then we utilize the Empirical Bayes method in the previous section to estimate the meta-prior variance for the first-order and second-order features τ_1^2 and τ_2^2 respectively. We restart the model with the new informative prior $\mathcal{N}(0, \tau_1^2)$ and $\mathcal{N}(0, \tau_2^2)$ for first-order and second-order features with the original data and update the model in batches.

Note that small t represents small traffic of observations. In our experiment, we've computed degenerative meta-prior variance, i.e. negative. This case shows the insufficiency of samples to estimate the meta-prior variance so that $\tilde{\mu_{i,1}}$ stay close to 0 and $\tilde{\sigma_{i,1}^2}$ stays close to 1. This leads to the negative τ_k^2 according to the equation (7). Hence, one approach to cope with this is to increment the samples by bootstrapping the observations behind the current time t to produce non-degenerative variance. In this simulation, we bootstrap the first batch into several batches (8 batches in our setting) and compute the variance through bootstrapped data.

BLIPTwice: This model is updated in batches twice, one with the bootstrapped data and another with the original observations, which aims at studying the effect of data reuse.

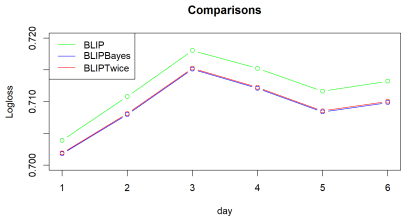
Finally, after training these three models, we evaluate the performance of the model through the cross entropy:

$$LogLoss = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log p(y_i = 1|x_i) + (1 - y_i) \log p(y_i = 0|x_i)] \tag{18}$$

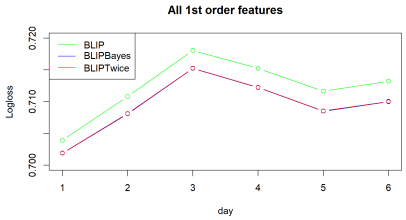
In the next subsection we will show the results of the experiments.

3.2 Results and Discussions

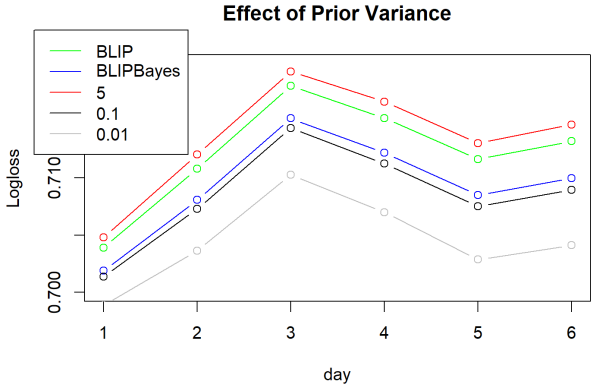
In this section we will manifest the performance of three proposed models and discuss the relavent extentions, including the reset time and prior variance settings, etc.



(a) All 1st



(b) Selected 1st



3.2.1 Effect of First-Order Features

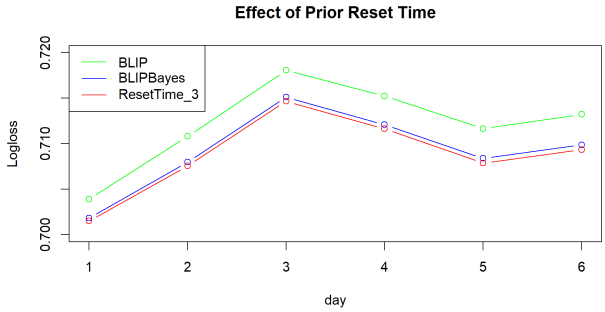
Since the adaptive LASSO prunes the data with incomplete first-order data and intuition that the first-order feature could reflect more information about the outcome, we retrain the model with adding the residual of the first-order features to the original data in three models. The logloss of three models with all first-order and selected first-order features are shown in the following figures.

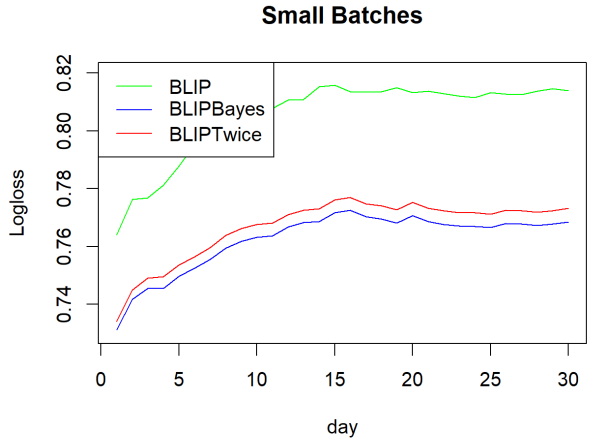
From the figures we could observe that the performance of three models with selected first-order data and all first-order data are similar, which represents that the adaptive LASSO has successfully selected the most important features that extracts the largest information towards the outcomes. Additionally, we could see the logloss of three models didn't have a monotonic decrease as the training process continues different from the results in **?** Nabi's paper. We speculate that the reason is that the update formula here utilize the method of data augmentation, which introduce the uncertainty during the training epoch so that the logloss increases holistically. Here still remains a doubt whether the authors adopted the component-wise update formula or not since we've analyzed the running speed of this method in R.

3.2.2 Effect of Prior Variance

Since we've studied the Empirical Bayes method of estimating the meta-prior variance, we could observe that the estimated meta-prior variance is less than 1, which represents the non-informative prior. Therefore, we may conduct a hypothesis that the lower the meta-prior variance, the better performance the model will obtain. Hence, we turn to study the effect of prior variance, where we experiment with additional meta-prior variance settings, i.e. $\tau_k^2 = 5, 0.1, 0.01$ respectively for each $k \in \{1, 2\}$. The figure plots the log loss for aformentioned scenarios.

From the figure we could verify our hypothesis that the lower the meta-prior variance, the better performance the model will obtain. However, the results is inconsistent to the results in **?**. We could observe the $\tau_k^2 = 5$ case under-performs $\tau_k^2 = 0.1$ case.





3.2.3 Effect of Reset Time

The reset time determines the traffic of the dataset when we train the Empirical Bayes BLIP. Here we set the reset time as $t = 3$ and compare with the $t = 1$ case of BLIP and BLIPBayes. The figure shows the logloss of them. We conclude that BLIPBayes still provides a lower log loss than BLIP. Additionally, BLIPBayes with reset time $t = 3$ possesses a better performance than BLIPBayes with reset time $t = 1$. This result indicates that the more data involved improves the performance.

3.2.4 Effect of the Size of Batches

In this experiment, we consider smaller size of batches for the model training and observe the perfomance of EB when the training period is longer. Here we set the size of batch as 1000, reset time at $t = 1$ but bootstrap for 12 times. Figure shows that BLIPBayes still better outperforms BLIP and BLIPTwice models than medium size of batches situations, which indicates BLIPBayes performance is consistent and valuable for small batching.

4 Conclusion

This paper aims to study the performance of empirical bayes method in estimating the category-specific informative prior with the early observed and bootstapped data instead of a non-informativ prior, which decouples the learning rate of different categories and provides a better performance. Additionally, the GLM-based empirical bayes estimator owns good properties of unbiaseness, strong consistency and an optimal $O(d^{\frac{3}{2}}\sqrt{T})$ regret bound. Empirical experiments reveal the higher cumulative rewards, better performance than single GLM estimator and lower convergence times for the Empirical Bayes techniques. However, the result of the simulation we conduct is relatiely different from the original paper's, especially, the non-decrease log loss of the model and the trade-off between high running speed satisfying the requirement of online learning and update strategy. Additoinally,

combining the Gaussian Imagination and the Empirical Bayes techniques to provide a more robust estimator is a promising approach.

5 Extensions

5.1 Overview of Empirical Bayes and Recent Researches

Empirical Bayes method, which is first proposed by Robbins **?** to cope with the following situation: a random sample of realizations $\theta_1, \theta_2, ..., \theta_N \sim g(\theta)$ which are unobserved but each realization has an observed random variable from a known family of distributions $f(x_i, |\theta_i)$:

$$\theta_i \sim g(\theta), \quad x_i \sim f(x_i|\theta_i) \tag{19}$$

Note that each x_i is drawn independently. Our goal is to estimate the parameter $\theta_1, \theta_2, ..., \theta_N \sim g(\theta)$.

Before Empirical Bayes, the most popular way to estimate the parameters is MLE, which represents that $\theta_i = f(x_i)$ for each i. However, Robbins' formula provides an insight that the other observation with respect to x_i (i.e. x_j where $j \neq i$) could be used to estimate θ_i. Hence, the estimation of the parameters becomes purely frequantistic, which is the main idea of Empirical Bayes. Robbins' formula, Missing Species problem, James-Stein estimator and False Discover Rate are the classical applications of the EB method that "Learning the experience from others".

According to whether we focus on estimating the prior density $g(\theta)$, the method in EB could branch into two mainstreams: f-modeling and g-modeling which means that we must make assumptions on either f or g. Most of modern EB methods assume a known f to construct the posterior mean without any assumption about the prior $g(\theta)$. Worthnoting, it's intractable to avoid making such strong assumptions about f. The Normal-Normal model could illustrate this dilemma. However, in the article of **?**, the authors proposed that the Normal-Normal model could be identifiable if there exists replicates for each unit i which developed a new path the estimate the posterior mean without any assumptions of f and g. This method is called Aurora aiming to match the risk of Bayes rule. The key insight is that the conditional mean $\mathbb{E}_{F,G}[Z_{ij}|X_i]$ is (almost surely) identical to the posterior mean $\mathbb{E}_{F,G}[\mu_i|X_i]$ with the property of the invariance of sufficiency of order statistics even in high-dimensional situations, which represents that Bayes rule could be estimated via $\mathbb{E}_{F,G}[\mu_i|X_i]$ and simply regress the splited single sample on the remaining ordered replicates under any black-box predictive model. Through averaging the estimated posterior mean for each replicate we could obtain the final answer while reducing the total variance. Additionally, Aurora manifests good properties of asymtotical bayes risks and universal consistency with KNN estimator.

Furthermore, researchers have dived into the point estimation problem in Empirical Bayes for long but neglect the importance of measuring the uncertainty of the estimators using EB methods where the confidence intervals are powerful tools to accomplish this **?**. The authors of Aurora also proprosed two methods of constructing confidence intervals for the marginal distribution of the observed random variables, F-localization and AMARI which build a simultaneous interval and pointwise interval respectively. Both two approches will finally be transformed into an optimization problem using the Charnes and Cooper transformation for linear-fractional programming, which provides a convenient way to apply optimization algorithms to compute the confidence intervals efficiently.

5.2 Historical Development of Bayesian Linear Probit Model

Bayesian Linear Probit model could stem from an approximation technique named Assumed Density Filtering **?**, which aims at seeking for an approximation density of the posterior $p(y|x)$ and could be viewed as a projection towards the expotential family via KL-divergence. Through derivations conclusion could be drawn that $\nabla_\theta KL(p||q) = 0 \Rightarrow \mathbb{E}_q[\Phi(y)] = \mathbb{E}_p[\Phi(y)]$, which represent that it's sufficient to match moments to minimize the KL-divergence. Also, if we assume that the posterior could be rewritten as a factorized distribution $p(y|x) = \prod_i t_i(y)$, ADF is still applicable to this case through matching moments for each factor.

However, the order of the factor to create approximation will change the final answer for the sake of linear step we go through the factor to approximate the posterior. Hence, another techique called expectation propagation is proposed to cope with the problem mentioned above, which update the approximation in a recursive way.

On the other side, with the aim of expressing the dependence of random variables in joint distribution decomposition and calculating the marginal distribution of arbitrary variables, a concept of factor graph is proposed, where the rectangle point, circle point and line represent the factors, variables and dependence between the factor and variable respectively. Further, based on the method of expectation propagation, a techique called belief propagation,which could be extracted to Sum-Product Algorithm, is proposed to calculate the marginal distribution of arbitrary variables in factor graph.

The brief derivation of the Sum-Product Algorithm **?** is as follows. Since the marginal distribution of an arbitrary node could be expressed as:

$$p(x) = \sum_{\mathbf{X}\backslash x} p(x) = \sum_{\mathbf{X}\backslash x}[\prod_{s\in ne(x)} F_x(x, X_s)] = \prod_{s\in ne(x)}[\sum_{X_s} F_x(x, X_s)] = \prod_{s\in ne(x)} \mu_{f_s\to x}(x) \quad (20)$$

where $ne(x)$ denotes the set of neighbors of node x and $F_x(x, X_s)$ denotes the factor f_s which is connected to node x. We rewrite it as $\mu_{f_s\to x}(x)$ and $\mu_{x_m\to f_s}(x)$ to represent the message from factor f_s to node x and the opposite respectively. These messages could calculated as follows:

$$\mu_{f_s\to x}(x) = \sum_{x_1} ... \sum_{x_M} f_s(x, x_1, ..., x_M) \prod_{m\in ne(f_s)\backslash x}[\sum_{X_{xm}} G_m(x_m, X_{sm})] \quad (21)$$

$$= \sum_{x_1} ... \sum_{x_M} f_s(x, x_1, ..., x_M) \prod_{m\in ne(f_s)\backslash x}[\mu_{x_m\to f_s}(x_m)] \quad (22)$$

$$\mu_{x_m\to f_s}(x_m) = \sum_{X_{sm}}[\prod_{l\in ne(x_m)\backslash f_s} F_l(x_m, X_{ml})] \quad (23)$$

$$= \prod_{l\in ne(x_m)\backslash f_s}[\sum_{X_{ml}} F_l(x_m, X_{ml})] \quad (24)$$

$$= \prod_{l\in ne(x_m)\backslash f_s} \mu_{f_l\to x_m}(x_m) \quad (25)$$

Combing the above formula, we could transform the marginal distribution as:

$$p(X_s) = f_s(x_s) \prod_{i\in ne(f_s)} \mu_{x_i\to f_s}(x_i) \quad (26)$$

Equipped with the above formula, we could also obtain the marginal distribution of all variables in factor graph in a reversed message passing way instead of applying complete algorithm to all variables which is time-consuming.

This techique captures the attentions from the field of player matching in computer games, where Microsoft proposed TrueSkill **??** algorithm to match close rivals through the prior estimation and update the skill level of players through factor graph and Sum-Product Algorithm. One of the specific cases is exactly the Bayesian Linear Probit model and its derivation is already mentioned in the previous section.

5.3 Relationship between MAML and Empirical Bayes

Rapid development of artificial intelligence has spectacular influence in multiple aspects such as computer vision, natural language processing and recommender system, etc. The generalization ability in multiple tasks instead of solving merely a specific problem is highly required for artificial general intelligence, which refers to the notion of meta-learning and imitates a remarkable aspect of human intelligence to quickly solve a novel problem in an unknown domain or domain with limited experience. The mechanism behind such fast adaptation is through leveraging prior and extract the domain-shared knowledge to accommodate to multiple types of tasks and improve the efficiency.

Tremendous researchers have developed different methods to construct the meta-learning framework. Two prevailing way of construction of meta-learning are the gradient-based hyperparameter optimization and probabilistic inference in a hierarchical Bayesian model.

We set MAML **?** as an example of the first construction method, which provides a gradient-based meta-learning procedure that employs a single additional parameter (the meta-learning rate) and operates on the same parameter space for both meta-learning and fast adaptation. The objective of MAML is as follows:

$$\mathcal{L}(\theta) = \frac{1}{\mathcal{J}} \sum_{j} [\frac{1}{\mathcal{M}} \sum_{m} -logp(x_{j_{N+m}} | \theta - \alpha \nabla_{\theta} \frac{1}{\mathcal{N}} \sum_{n} -logp(x_{j_N} | \theta))] \tag{27}$$

An alternative way to formulate meta-learning is as a problem of probabilistic inference in the hierarchical model. The insight of the applying the hierarchical bayesian model is that the task-specific parameters will influence the estimation of other tasks, represented by introducing a meta-level parameter to encode the statistical dependence of task-specific parameters. The objective of this method is to maximize:

$$p(X|\theta) = \prod_{j} (\int p(x_{j_1}, ..., x_{j_N} | \theta_j) p(\phi_j | \theta) d\phi_j) \tag{28}$$

as a function of θ to obtain a point estimation, an instance of empirical bayes which is a suitable approach to estimate the prior parameters.

? provides a perspective towards the connection between gradient-based meta-learning and hierarchical empirical bayes, showing that MAML could be interpreted as a hierarchical probabilistic model. The following is the derivation of the link.

From equation (17), obtaining the exact marginal distribution of task-specific parameter ϕ_j in this objective is intractable and time-consuming. Hence, considering an approximation of the negative log-likelihood with the arbitrary point estimator $\hat{\phi}_j$:

$$- logp(x|\theta) \approx \sum_{j} [-logp(x_{j_{N+1}}, ..., x_{j_{N+M}} | \hat{\phi}_j)] \tag{29}$$

If we set $\hat{\phi}_j = \theta + \alpha \nabla_{\theta} logp(x_{j_1}, ..., x_{j_N} | \theta)$ for each j, we could obtain the unscaled form of objective function of MAML with the method of gradient-based hyperparameter optimization, which represents that the MAML objective is equivalent to a maximizationi with regard to a meta-level paremeter. In fact, this equivalence also manifest the task-specific parameter's trade-off between staying close to the meta-level parameter in sense of specific norm and minimizing the fast adaptation, i.e. the early stopping in fast adaptation is equivalent to the specific setting of the task-specific parameter conditional on the meta-level parameter $p(\phi_j | \theta)$.

We could illustrate this conclusion in the following case. Considering the second-order approximation of fast adaptation $\ell(\phi) \approx \tilde{\ell}(\phi) := \frac{1}{2} \left\| \phi - \phi^* \right\|^2_{\mathbf{H}^{-1}} + \ell(\phi^*)$, where the Hessian could be extended to curvature matrix for gradient in gradient descent and derive the update formula as $\phi_{(k)} = \phi_{(k-1)} - \mathcal{B} \nabla_{\phi} \tilde{\ell} \left(\phi_{(k-1)} \right)$. Specifically, the meta-learned curvature matrix incorperate the task-general information into the covariance of the fast adaptation so that reflects the interaction between task-specific parameters. According to the work of **?**, the objective of updating the task-specific parameter is formalized as:

$$min(||\phi - \phi^*||^2_{H^{-1}} + ||\phi_{(0)} - \phi||^2_Q) \tag{30}$$

which is equivalent to choose Gaussian prior with specific mean θ and covariance $Q = O\Lambda^{-1}((I - B\Lambda)^{-k} - I)O^T$, where B is a diagonal matrix that results from a simultaneous diagonalization of H and B as $O^T H O = diag(\lambda_1, ..., \lambda_n) = \Lambda$ and $O^T B O = diag(b_1, ..., b_n) = B$.

5.4 Bayesian Method in Multi-Armed Bandits and Recommender System

Lots of scenarios in recommender system will encounter the situation where the agent seeks for the best choice in multiple options that represents the consumers' previous preferences and potential interests. Another example is the cold-start problem to conject the customers' rough preferences who didn't browse the apps for a relatively long period. These could be extracted as the exploration-exploitation trade-off and Multi-Armed Bandits problem.

Mainstream methods in multi-armed bandits are splited into non-Bayesian method, e.g. ϵ-greedy, UCB, etc, and Bayesian ones such as Thompson sampling, etc. Here we provide a brief introduction of Bayesian methods in multi-armed bandits, Thompson sampling and its variants.

The original TS method assumes that probability of obtaining reward from each arm is a Beta distribution with parameters α and β, which could be interpreted as the times of winning and losing respectively. We choose one arm to play and update the parameters of Beta distribution after each round according to following strategies: α is updated by $\alpha + 1$ if received reward and β is updated by $\beta + 1$ if received no reward. The strategy we choose the arm is to sample the current Beta distribution of all arms and select the arm with the largest sample. The intuition of TS sampling is the meanings of the parameters of Beta distribution. It's also noted that Beta distribution is conjugate to the Bernoulli distribution which is exactly the density of the arms' reward.

Original TS method is a simple and effective way to deal with the E-E problem. However, in real-world applications, we have limited knowledge about the prior of arms, i.e. the probability of gaining reward from certain arm may not Beta distributed or furthermore not conjugate to bandits' settings. Recent research **?** has proved that TS sampling is still effective even if the distribution behind the prior and likelihood fucntion is Gaussian but the bandits' settings are still Bernoulli. In other words, while interacting with the real environment that is Bernoulli distributed, we consider the observed data as Gaussian distributed and update the parameters of Gaussian distribution for each round, which is called "Gaussian Imagination". The authors proved that under the following two assumptions: For all $t \in \mathbb{Z}_{++}$, $\mathbb{E}\left[\mathbb{E}\left[\tilde{R}_* \mid \tilde{H}_t \leftarrow H_t\right]\right] \geq \mathbb{E}\left[R_*\right]$; The imaginary learning target $\tilde{\chi}$ and the imaginary mean reward $\tilde{\theta}$ are jointly Gaussian. The regret bound is upper bounded as follows:

$$\mathcal{R}(T) \leq \sqrt{\mathbb{I}(\tilde{\chi}, \tilde{\mathcal{E}})\tilde{\Gamma}_{\tilde{\chi},\epsilon} T} + \epsilon T + \gamma\sqrt{2 d_{KL}(\mathbb{P}(\theta \in \cdot)||\mathbb{P}(\tilde{\theta} \in \cdot))T} \tag{31}$$

Comparing to the regret bound that the prior distribution and likelihood are conjugate to the bandits' settings:

$$\mathcal{R}(T) \leq \sqrt{\mathbb{I}(\chi, \mathcal{E})\Gamma_{\chi,\epsilon} T} + \epsilon T \tag{32}$$

for all learning target χ, tolerence $\epsilon \in \mathbb{R}_+$ and time horizon $T \in \mathbb{Z}_{++}$.

For sure, the misspecification of the prior distribution will lead to additional regret for each round. However, the regret is still upper bounded through change-of-measure approach with decomposition of the regret bound and this Gaussian bandit guarantees high degree of robustness to misspecification with a sufficiently diffuse prior distribution and likelihood function.

This provides a perspective that in certain sense, we could neglect the misspecification and approximate the best strategy in E-E dilemma. However, the extension to the misspecification of other familiy of densities still awaits. To my perspective, the optimization view mentioned in meta-learning is a novel question to answer which could help apply the optimization algorithm to the Multi-Armed Bandit problem.

References

T. Graepel, J. Quiñonero Candela, T. Borchert, and R. Herbrich, "Web-scale bayesian click-through rate prediction for sponsored search advertising in microsoft's bing search engine," in *Proceedings of the 27th International Conference on Machine Learning ICML 2010, Invited Applications Track (unreviewed, to appear)*, June 2010.

X. He, J. Pan, O. Jin, T. Xu, B. Liu, T. Xu, Y. Shi, A. Atallah, R. Herbrich, S. Bowers, and J. Q. n. Candela, "Practical lessons from predicting clicks on ads at facebook," in *Association for Computing Machinery*. Association for Computing Machinery, 2014.

J. H. Albert and S. Chib, "Bayesian analysis of binary and polychotomous response data," *Journal of the American Statistical Association*, vol. 88, no. 422, pp. 669–679, 1993.

S. Nabi, H. Nassif, J. Hong, H. Mamani, and G. Imbens, "Bayesian meta-prior learning using empirical bayes," 2020.

B. Efron, "Empirical bayes: Concepts and methods," 2010. [Online]. Available: https://efron.ckirby.su.domains/papers/2021EB-concepts-methods.pdf

N. Ignatiadis, S. Saha, D. L. Sun, and O. Muralidharan, "Empirical bayes mean estimation with nonparametric errors via order statistic regression on replicated data," 2019.

N. Ignatiadis and S. Wager, "Confidence intervals for nonparametric empirical bayes analysis," 2019.

D. Khashabi, "Expectation propagation for bayesian inference," 2010. [Online]. Available: https://danielkhashabi.com/learn/ep.pdf

C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4, no. 4.

R. Herbrich, T. Minka, and T. Graepel, "Trueskill™: A bayesian skill rating system," in *Advances in Neural Information Processing Systems*, B. Schölkopf, J. Platt, and T. Hoffman, Eds., vol. 19. MIT Press, 2006.

F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.

E. Grant, C. Finn, S. Levine, T. Darrell, and T. Griffiths, "Recasting gradient-based meta-learning as hierarchical bayes," 2018.

Y. Liu, A. M. Devraj, B. Van Roy, and K. Xu, "Gaussian imagination in bandit learning," 2022.