Perovskite Model Description

August 14, 2017

1 Introduction

In this project we are interested in finding a perovskite composite with the highest unsaturated Mayer Bond order (UMBO). We approach this problem with Bayesian Optimization. This documentation provides model description of the single-halide scenario and presents results obtained by using the model.

2 Single-halide Model

2.1 Model setup

In a single-halide model, the setup contains one halide, one cation and one solvent. There are 3 choices for halides, 3 for cations and 8 for solvents, hence 72 possible setups in total. Let V_x to denote the value (UMBO) of setup x, $x \in \{1, ..., 72\}$. We use vector Z^x to describe the components of setup x, with each index i corresponding to a possible element in the setup. In particular, $Z_i^x, i \in \{1, 2, 3\}$ is a binary variable that indicates whether a cation is present in setup x. We require that $\sum_{i=1}^{3} Z_i^x = 1, \forall x$, since there is only one cation present in each setup. Likewise, $Z_i^x, i \in \{4, 5, 6\}$ is a binary variable that indicate whether a halide is present in setup x. $Z_7^x \in \{s_1, ..., s_8\}$ indicates which solvent is used in setup x.

For example, $Z^x = (1, 0, 0, 0, 1, 0, s_4)$ means that setup x contains the first type of cation, second type of halide, and the 4^{th} type of solvent. In addition, for future computation, we let s_i be a 2-d vector with the first entry being the MVEE and the second entry being the polarity of the i^{th} solvent, $i \in \{1, ..., 8\}$.

We assume the presence of cations and halides contributes linearly to the objective value, and we use α_i to quantify the amount contributed by cation/halide i. To correct for the possible non-linear effects of cations and halides, we introduce β_x for each of the setup x. For solvents, we assume they affect the solubility through an unknown function $f : \mathbb{R}^2 \mapsto \mathbb{R}$ which takes into consideration the MVEE and polarity of a solvent. In addition to just a halide, a cation and a solvent, a perovskite also contains a central ion, for which we choose lead for all compositions. We use ζ to denote the contribution of the central ion to the objective function. Hence solubility of setup x can be written in the following way:

$$V_x = \sum_{i=1}^{6} \alpha_i Z_i^x + \beta_x + \zeta + f(Z_7^x). \tag{1}$$

We place the same prior distribution $N(\mu_\alpha, \sigma_\alpha^2)$ on each of α_i, and $N(0, \sigma_\beta^2)$ on β_x. ζ is assume to follow $N(\mu_\zeta, \sigma_\zeta^2)$ According to the Bayesian approach we suppose that $f(\cdot)$ is drawn from a Gaussian process with prior mean function $\mu_0(\cdot)$ and covariance $\Sigma_0(\cdot, \cdot)$. Since ζ already captures the invariant contribution from the central ion, it is reasonable for us to assume $\mu_0(\cdot) = \mathbf{0}$. We assume that two setups will have similar objective values if they differ little in their parts. To formalize this, for $\Sigma_0(\cdot, \cdot)$, we use 5/2 Matérn kernel. Let $r = \sqrt{\sum_{i=1}^{d} \ell_i (x_{1,i} - x_{2,i})^2}$ denote the distance between point x_1 and x_2 weighed by each dimension. In our case $d = 2$. The covariance between x_1 and x_2 under a 5/2 Matérn kernel is calculated as:

$$\Sigma_0(x_1, x_2) = \sigma_m^2 \left(1 + \sqrt{5}r + \frac{1}{3}5r^2\right) exp\left(-\sqrt{5}r\right), \tag{2}$$

where σ_m^2, ℓ_i are hyper-parameters. With all the components of V_x being normally distributed, V_x is also normally distributed. Hence we are able to describe the joint distribution of $(v_1, ..., v_{27})$ with a multivariate normal distribution:

$$\begin{pmatrix} V_1 \\ V_2 \\ \vdots \\ V_{135} \end{pmatrix} \sim N(\mu^0, \Sigma^0), \tag{3}$$

where

$$\begin{aligned} \mu_x^0 &= \mathbb{E}\left[\sum_{i=1}^{6} \alpha_i Z_i^x + \beta_x + \zeta + f(Z_7^x)\right] \\ &= \sum_{i=1}^{6} \mu_\alpha Z_i^x + \mu_\zeta + \sum_{i=1}^{15} \mathbb{1}(Z_7^x = s_i)\mu_{0,i} \\ &= 2\mu_\alpha + \mu_\zeta, \end{aligned}$$

and

$$\begin{aligned} \Sigma_{x,x'}^0 &= COV(V_x, V_{x'}) \\ &= \sum_{i=1}^{6} \sigma_\alpha^2 \mathbb{1}(Z_i^x = 1)\mathbb{1}(Z_i^{x'} = 1) + \Sigma_0(Z_7^x, Z_7^{x'}) + \mathbb{1}(x = x')\sigma_\beta^2 + \sigma_\zeta^2 \end{aligned}$$

2.2 Estimation of Hyper-parameters

We estimate the hyper-parameters by maximum likelihood estimation method. Let $\theta = \{\mu_\alpha, \sigma_\alpha, \sigma_\beta, \mu_\zeta, \sigma_\zeta, \ell_1, \ell_2\}$ be the vector of all the hyper-parameters we want to estimate. Let $V = \{v_{i_1}, ..., v_{i_m}\}$ to be the observations we made for $i_1, ..., i_m$. Then the likelihood of V given θ is

$$\begin{aligned} L(\theta) &= P(V|\theta) \\ &= (2\pi)^{-m/2} * |\Sigma^0(i_1, ..., i_m)| * \exp\left(-\frac{1}{2}u^0(i_1, ..., i_m)^T \Sigma^0(i_1, ..., i_m) u^0(i_1, ..., i_m)\right), \end{aligned}$$

where $u^0(i_1, ..., i_m) = (u^0_{i_1}, ..., u^0_{i_m})$ is a vertical vector, and $\Sigma^0(i_1, ..., i_m)u^0(i_1, ..., i_m)$ is a $m \times m$ matrix with $\{k, j\}$ entry $= \Sigma^0_{i_k, i_j}$.

The MLE of θ is set to be

$$\hat{\theta} = \underset{\theta}{\text{argmax}}\ log(L(\theta)). \tag{4}$$

A maximizer of (4) can be found via gradient-based optimization algorithms such as BFGS.

2.3 Posterior update

We can update the mean and covariance in (3) after we make an observation. Let μ^n and Σ^n denote the mean and covariance in (3) after n observations are made. It is a well known fact that we can update the posterior distribution in the following way: given μ^n and Σ^n, and a new observation $\hat{y}^{n+1}$ which is made about setup x, the new posterior is:

$$\mu^{n+1} = \mu^n + \frac{\hat{y}^{n+1} - \mu^n_x}{\Sigma^n_{xx}} \Sigma^n \mathbf{e}_x, \tag{5}$$

$$\Sigma^{n+1} = \Sigma^n - \frac{\Sigma^n \mathbf{e}_x \mathbf{e}_x^T \Sigma^n}{\Sigma^n_{xx}} \tag{6}$$

2.4 Choosing the next composition to be studied

Expected Improvement (EI) is used to determine which setup x the next observation is to be made. Let $\hat{y}^* = \max_{1,...,n}\{\hat{y_1}, ..., \hat{y_n}\}$ be the largest observed value so far. Assuming a maximizing problem, the expected improvement (EI) of a setup x is:

$$\begin{aligned} EI(x) &= \mathbb{E}\Big[\max\{V_x - \hat{y}^*, 0\}\Big|\mu^n, \Sigma^n\Big] \\ &= (\mu^n_x - \hat{y}^*)\Phi\left(\frac{\mu^n_x - \hat{y}^*}{(\Sigma^n_{xx})^{1/2}}\right) + (\Sigma^n_{xx})^{1/2}\phi\left(\frac{\mu^n_x - \hat{y}^*}{(\Sigma^n_{xx})^{1/2}}\right), \end{aligned}$$

where $\Phi(\cdot)$ and $\phi(\cdot)$ are standard normal cdf and pdf respectively.

We pick the setup with the largest EI:

$$x^* = \underset{x \in \{1,...,72\}}{\text{argmax}} \; EI(x) \tag{7}$$

2.5 Experiment Result

We first use leave-one-out cross validation to verify that our model fits the data well. 20 samples (compositions and their values) are selected randomly. Each time one sample is left out and hyperparameters of our model are estimated using the rest 19 samples. We then compare the values of the sample left out to the confidence interval of the value calculated based on the parameters estimated. Figure 1 plots the result of the cross validation. It can been seen that most of the observed value lies within the corresponding confidence interval, thus indicating a good fit of the model to the data.

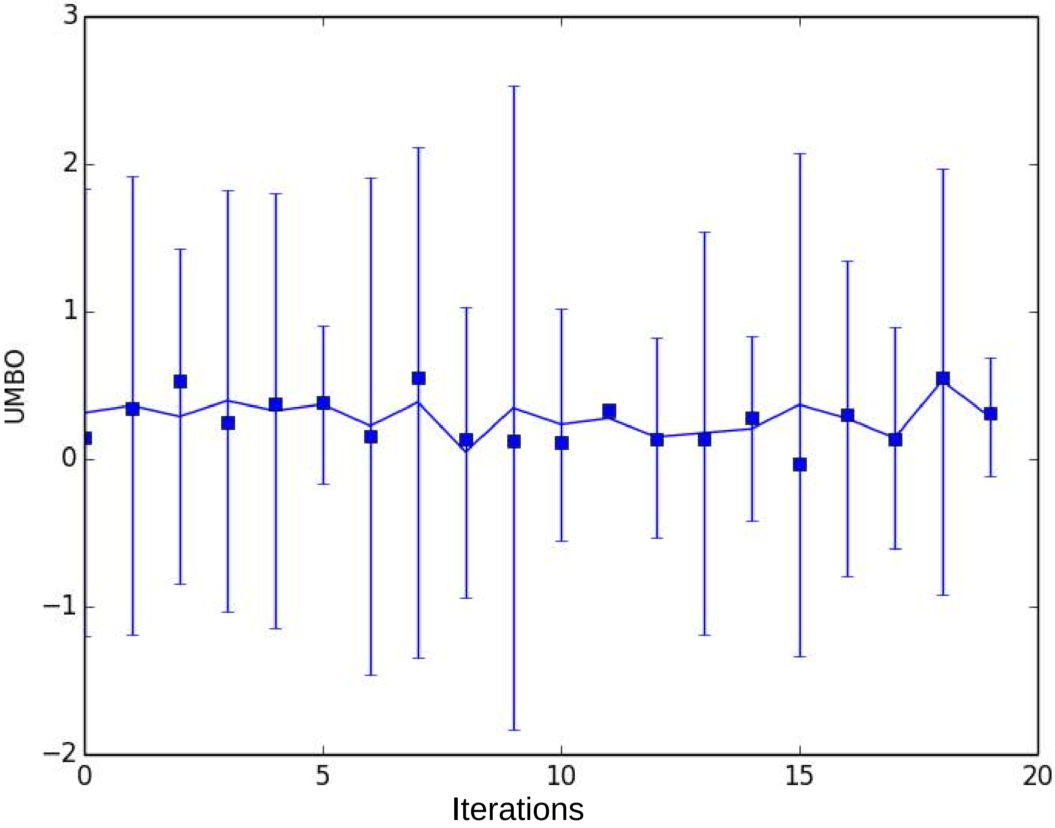

Figure 1: Plot of results of leave-one-out cross validation for 20 samples

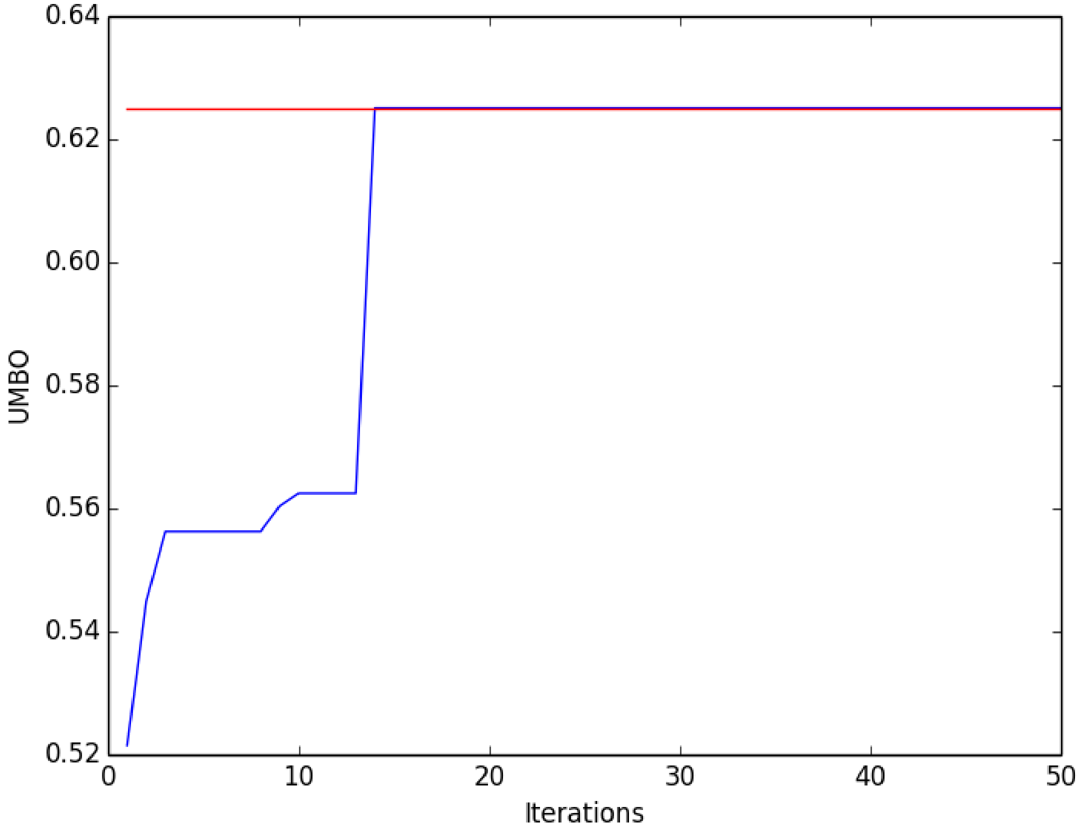

Figure 2: Plot of the the maximum values by iteration

Figure 2 plots the result of the sampling process by using the Bayesian optimization method aforementioned. The red line shows the true maximum value among all the possible setups. We can see that it takes 14 iterations for the BO process to converge to the true maximum, which is considerably less effort than carrying out the sampling randomly.

3 Mixed-Halide Model

3.1 Model setup

Let V_x to denote the value of setup x. For each setup, there are three halides, one cation and a solvent. Since the positioning of the halides matters, there are in total 27*3*8 = 648 possible combinations. (i.e., $x \in \{1, ..., 648\}$). We use vector Z^x to describe the components of setup x. In particular, $Z_i^x, i \in \{1,2,3\}$ is a binary variable that indicates which halide is in the 1st position. We require that $\sum_{i=1}^{3} Z_i^x = 1, \forall x$. For example, $Z_{1,2,3}^x = (0,0,1)$ means that iodide is present in the 1st position. Likewise, $Z_i^x, i \in \{4,5,6\}$ is a binary variable that indicates which halide is present in the second position and $Z_i^x, i \in \{7,8,9\}$ in indicates which halide is present in the third position. Z_i^x, $i \in \{10,11,12\}$, indicates which cation is present in the setup. Again it is required that $\sum_{i=10}^{12} Z_i^x = 1, \forall x$. $Z_{13}^x \in \{s_1, ..., s_8\}$ indicates which solvent is used in setup x. In addition, for future computation, we let s_i be a 2-d vector with the first entry being the MVEE and the second entry being the polarity of the i^{th} solvent, $i \in \{1, ..., 8\}$.

We still assume the presence of cations and halides contributes linearly to the value, and we use α_i to quantify the amount contributed by cation/halide i. To correct for the possible non-linear effect of cations and halides, we introduce β_x for each of the setup x. For solvents, we assume they affect the UMBO through an unknown function $f : \mathbb{R}^2 \mapsto \mathbb{R}$ which takes into consideration the MVEE and polarity of a solvent. Moreover, since a perovskite setup contains more than just a cation, a halide and a solvent, we use ζ to denote the invariant part of the setup. Hence UMBO of setup x can be written in the following way:

$$V_x = \sum_{i=1}^{12} \alpha_i Z_i^x + \beta_x + \zeta + f(Z_{13}^x). \tag{8}$$

We place the same prior distribution $N(\mu_\alpha, \sigma_\alpha^2)$ on each of α_i, and $N(0, \sigma_\beta^2)$ on β_x. ζ is assume to follow $N(\mu_\zeta, \sigma_\zeta^2)$. We also view $f(\cdot)$ as a Gaussian process with prior $\mu_0(\cdot)$ and $\Sigma_0(\cdot,\cdot)$. Since ζ captures the invariant part of the setup, and somewhat a fixed amount to the solubility value, it is reasonable for us to assume $\mu_0(\cdot) = \mathbf{0}$. For $\Sigma_0(\cdot,\cdot)$, we use 5/2 Matern kernel. Let $r = \sqrt{\sum_{i=1}^{d} \ell_i (x_{1,i} - x_{2,i})^2}$ denote the distance between point x_1 and x_2 weighed by each dimension. In our case $d = 2$. The covariance between x_1 and x_2 under a 5/2 Matern kernel which is the same as the one described in section 1.

We place the same distributions on the parameters as in Section 1. We also use the same procedure described in section 1 to obtain prior parameters and to update posterior.

3.2 single-sample batch

For single-sample batch, we select one point to sample in every batch. The computation of EI and selection criterion for the next batch are the same as what has been described in section 2.4.

3.3 multi-sample batch

For multi-sample batch, we select I points to sample in every batch. To be more precise, the batch-EI of I points $x_1, ..., x_I$ is defined as follows:

$$EI(x_1, ..., x_I) = \mathbb{E}\left[(\max_{i \in \{1,...,I\}} V_i - f*)^+ \middle| \mu_n, \Sigma_n \right] \tag{9}$$

An exact computation of the batch-EI is given in the paper *Fast Computation of the Multi-points Expected Improvement with Applications in Batch Selection* by Chevalier and Ginsbourger (https://hal.archives-ouvertes.fr/hal-00732512/document). Let b be the number of samples in a batch. We use the following algorithm to find the set $B = \text{argmax}_{B \subseteq 1,...,N} EI(B)$, with the cardinality of B being b. (Recall N is the total number of solutions.): The rest of the computation follows the same as the single-halide model.

Algorithm 1 Algorithm for selecting B sequentially

```
$B := \{\}$
$set = \{x_1, ..., x_N\}$
**for** $i \in \{1, ..., b\}$ **do**
    compute $x^* = \text{argmax}_{i \in \{1,...,N\}} EI(B \cup \{x_i\})$
    $B = batch \cup \{x^*\}$
    $set = set \backslash \{x^*\}$
**end for**
```