Bayesian Optimization of a Molecular Simulation of H_2O

Harry Winston Sullivan, Leqian(Tim) Tan

1 Executive Summary

Our project aims to answer *what is the best classical molecular model of water which can reproduce experimental scattering measurements?* More specifically, we want find the simplest molecular model which fits the oxygen-oxygen radial distribution function (RDF) $g_{OO}(r)$. Our understanding of the liquid state relies heavily on established theoretical relationships that link the RDF to thermodynamic properties and interatomic forces, therefore the RDF serves as a primary target for the optimization of molecular models broadly. To achieve this we applied the standard Bayesian optimization and uncertainty quantification framework.

Bayesian methods typically involve thousands of evaluations of a given model, which in our case implies one molecular dynamics (MD) simulation per evaluation. To reduce the computational bottleneck we trained a Local Gaussian Process (LGP) machine learning model which can efficiently be evaluated in an embarrassingly parallel linear scaling $O(\eta)$ in the number of experimental structure observations η. Our training and testing dataset consisted of $N = 512$ MD simulations generated using LAMMPS. From simulation trajectories we computed the oxygen-oxygen RDF, finally giving the dataset generated data set as N tuples of the form $\{\phi,g_{OO}(r;\phi)\}$ where ϕ is a molecular model. Utilizing a leave-one-out N-fold cross validation algorithm we fit the hyperparameters of the LGP model by maximizing the posterior probability of the training dataset.

With a trained LGP model we then performed inference of the molecular model parameters using Hamiltonian Monte Carlo (HMC). Our results demonstrate a robust LGP model can serve as an efficient MD replacement. The HMC optimized parameters yield an improved liquid structure prediction compared to the most popular water model TIP4P/2005. Additionally the Bayesian optimization scheme computed the confidence intervals on the parameters and RDF predictions, supporting the validity of the learning procedure.

2 Background

The description of potential energy surfaces and their corresponding emergent thermodynamic properties is a central focus of liquid state theory. Recent studies of machine learning interatomic potentials (MLIP) for fluid systems, such as water or electrolyte solutions, have shown low transferability to different thermodynamic conditions and high deviations in energy predictions in out-of-sample regions of phase space. Authors have explained this deviation by the lack of a correct description of the liquid structure $g_{OO}(r)$ [1, 2]. This could be due to both approximation error of the MLIP or underlying errors in the DFT functional the data was generated with. In principle this could be resolved by employing a higher level of theory (coupled cluster, path integral MD, etc.). However, the limitation of computational cost constrains the size of the system we can include in training sets to scales where the radial distribution function (RDF) can't be computed. Therefore, it is essential to find an interatomic potential which can be trained quickly and still provides meaningful insight to the RDF.

Rather than apply deeper theory many studies exclusively utilize empirical molecular models to estimate the RDF as well as other thermodynamic properties. Among them, X-ray diffraction serves as a critical reference technique for probing the structural models of liquid water. Skinner and coworkers characterized the oxygen-oxygen RDF of water via an experimentally derived structure factor related to the RDF by a radial Fourier transform[3]. Despite the availability of this metric, most models fail to produce results that match observations not directly optimized against [4]. For

example, in the TIP4P/2005 paper[5] they fit broadly to phase behavior. While they have reached good agreement with the optimized against experimental data they failed to align with RDF. This is concerning for a few reasons, namely the Kirkwood-Buff relations [6, 7] connecting the RDF with the thermodynamic properties through integral equations. In principle if the RDF is fit well then the corresponding thermodynamic properties must be accurate. This begs the question, is the lack of simultaneous fitting of $g(r)$ and other thermodynamic properties due experimental uncertainty, model uncertainty, or a poor choice of optimization target? A key tool in data science to answer such questions is Bayesian inference. Bayesian inference can simultaneously optimize parameters as well as estimate their uncertainty, however typical techniques require many calls to the underlying model. To resolve this the introduction of a surrogate model [8] as well as highly efficient Hamiltonian Monte Carlo [9] can make this analysis feasible. The surrogate model can be almost any machine learning architecture, a typical Bayesian choice is the Gaussian process due to their closed posterior form as well as efficient hyper parameter optimization[10].

We study a rigid model of H_2O, assuming fixed bond angles and lengths, with variable intermolecular interactions. Following the assumptions of Abascal and Vega [5], we model the oxygen–oxygen (OO) interaction as pairwise and purely Lennard-Jones [11], while hydrogen atoms interact only via electrostatics. The oxygen's partial charge is displaced toward the hydrogen along the bisector. This defines our optimizable parameter set as $\phi = \{\epsilon, \sigma, q_{\mathrm{H}}, r_{\mathrm{OM}}\}$, representing the OO Lennard-Jones well depth, OO particle diameter, hydrogen partial charge, and the displacement of the oxygen charge toward the hydrogen atoms, respectively.

3 Data Description

Simulation data were generated using LAMMPS [12]. Guided by chemical intuition, we constrained the four-dimensional parameter space to a physically meaningful hypercube $\mathcal{A}$. The bounds defining $\mathcal{A}$ were: $\epsilon_{Lo}, \epsilon_{Hi} = 0.0717, 0.239$ kcal/mol; $\sigma_{Lo}, \sigma_{Hi} = 3.00, 3.40$ Å; $q_{\mathrm{H},Lo}, q_{\mathrm{H},Hi} = 0.4, 0.7$ e, and $r_{\mathrm{OM},Lo}, r_{\mathrm{OM},Hi} = 0.1, 0.2$ Å. Within $\mathcal{A}$, we used Sobol sampling [13] to select training points.[1] Simulation data were generated iteratively by shrinking the parameter domain by 5% over four successive rounds, concentrating computational effort near regions that better matched experimental structure. At each stage, the center of the domain was updated by minimizing the sum of squared errors in the OO RDF between simulation outputs and experimental data from Skinner [3]. We performed a total of four iterations and designated the final one as the test set, as it likely contained more physically plausible parameterizations. Since the surrogate model is primarily required to be accurate within the 95% confidence region of the posterior, which concentrates near the refined center, this strategy ensures modeling accuracy where it matters most.

A total of 512 simulations and their corresponding RDFs were generated for the training and test sets. Simulations were run in the NVT ensemble using a Nosé–Hoover thermostat with a 1 fs time step. Simulation details can be found in the SI template LAMMPS script. Preliminary runs at the 16 corners of $\mathcal{A}$ indicated that equilibration was achieved after approximately 4000 steps. These corners represent the extrema of the parameter space and are expected to exhibit the most pathological behavior within $\mathcal{A}$. By ensuring these points are sufficiently equilibrated we have confidence this also applies to the interior of $\mathcal{A}$. We double 4000 to 8000 for assurance this is the case. 20,000 production steps with sampling a stride of 50 was chosen to ensure samples of the first coordination number were uncorrelated.[2] This was determined via an autocorrelation function analysis. The integrated

[1] Sobol sampling is a quasi-random, low-discrepancy method that provides more uniform coverage than purely random sampling.

[2] We used the coordination number rather than the RDF as it represents a good summary metric of the RDF amenable

autocorrelation time was found to be 52.42 fs. By ensuring the coordination number samples are uncorrelated the RDF is likely to be as well. The RDFs were then computed using MDAnalysis[14]. Summary datum of the dataset is shown in figure S1 in SI. Exact sizes of the resulting dataset are provided in the methods section for flow purposes.

4 Methods

Inference can be described by Bayes theorem [15]. In a continuous parameter space this is given by the distribution function

$$p(\phi|\mathcal{D}) = \frac{\overbrace{p(\mathcal{D}|\phi)}^{\text{Likelihood}} \overbrace{p(\phi)}^{\text{Prior}}}{\underbrace{p(\mathcal{D})}_{\text{Evidence}}} = \frac{p(\mathcal{D}|\phi)p(\phi)}{\int d\phi\ p(\mathcal{D}|\phi)p(\phi)} \tag{1}$$

where ϕ is the set of molecular model parameters and $\mathcal{D}$ is the data. In order to answer the question posed in the the summary we choose $\mathcal{D} = \{g_{OO}(r_1), g_{OO}(r_2), \dots g_{OO}(r_\eta)\}$ and $\phi = \{\epsilon, \sigma, q_{\mathrm{H}}, r_{\mathrm{OM}}\}$ as determined by Skinner [3]. We also assume that the error in the experimental data is normally distributed, this implies

$$p(\mathcal{D}|\phi) = (2\pi\alpha)^{-\eta/2} \exp\left(-(2\alpha)^{-1/2} \sum_{i=1}^{\eta} (g_{OO}(r_i) - M(r_i, \phi))^2\right) \tag{2}$$

where α is the unknown observation noise of g_{OO}. We then assume the model parameters lie within a hyper-cubic domain $\mathcal{A}$. This implies that the parameters each satisfy an inequality $\phi \in \mathcal{A} \implies \phi_{Lo}^{(i)} \leq \phi^{(i)} \leq \phi_{Hi}^{(i)}$ where the superscript (i) denotes a specific component of ϕ in the parameter vector space. This allows us to restrict our posterior to a support that satisfies chemical intuition. Furthermore we assume a nearly uniform prior distribution $p(\phi)$ over $\mathcal{A}$.

The process of inference is then done via plotting equation 1 as a function of ϕ and evaluating expectation values of mappings f. Some examples of f could be moments (ϕ, ϕ^2, ..., ϕ^p), structure ($g_{OO}(r;\phi)$), and thermodynamical properties (heat capacity $C_V(\phi)$). This is in stark contrast to typical maximum likelihood or simple optimizer techniques which can not quantify the uncertainty in ϕ. This is not as straightforward as it seems for a few reasons.

- The posterior over the parameters is 4 dimensional. This implies we need a $4 + 2$ dimensional visualization technique to show equation 1 as a function of ϕ and α. While methods do exists they are all lackluster. Therefore we resolve this via marginalization of the posterior to only look at distributions over one or two parameters at the same time.
- The evidence term in 1 is intractable. This means we cannot evaluate the integral after the rightmost equal sign analytically. To resolve this we apply Markov Chain Monte Carlo to approximate 1 via a histogram as well as compute expected values of mappings f with $\frac{1}{n}\sum_{i=1}^{n} f(\phi_i)$.
- Each evaluation of equation 2 requires at least one evaluation of the model $M(r_i, \phi)$. This is extremely costly due to our model being a molecular dynamics simulation. Therefore we opt to find an approximation of the mapping $M : \mathcal{A} \to \mathbb{R}_0^+$. A local Gaussian process machine learning model is our surrogate approximation of choice.

For the following sections assume the training set is organized into an input matrix, X, of size (N

to simple time series analysis.

$\times$ dim(ϕ)) where N is the number of training simulations (512) and dim(ϕ) = 4. Also assume the training set output matrix, Y, is of size $(N \times \eta)$.

$$X = \begin{bmatrix} \phi_1^{(1)} & \phi_1^{(2)} & \phi_1^{(3)} & \phi_1^{(4)} \\ \vdots & \vdots & \vdots & \vdots \\ \phi_N^{(1)} & \phi_N^{(2)} & \phi_N^{(3)} & \phi_N^{(4)} \end{bmatrix}, \quad Y = \begin{bmatrix} g_{OO,1}(r_1) & g_{OO,1}(r_2) & \cdots & g_{OO,1}(r_\eta) \\ \vdots & \vdots & \ddots & \vdots \\ g_{OO,N}(r_1) & g_{OO,N}(r_2) & \cdots & g_{OO,N}(r_\eta) \end{bmatrix} \tag{3}$$

Each row of X corresponds to a set of simulation parameters which is then mapped to a vector of RDF values corresponding to a row in Y. The value η is meant to correspond to the number of experimental observations.

4.1 Local Gaussian Processes

A local Gaussian process is a special case of a standard Gaussian process (GP) which admits embarrassingly parallel linear scaling $O(\eta)$[16]. This scheme admits computational advantages in comparison to standard GPs. The amount of training data can be reduced significantly and the time complexity of the inversion of the kernel data matrix $K(X, X)$ quick. An LGP approximation can be described by the following equation: $\mathbb{E}[\mathcal{GP}_i](\phi) \approx M(r_i, \phi)$. This equation is meant to communicate that we are approximating the mapping from ϕ to $g_{OO}(r_i)$ at fixed r_i as the expectation of a GP. This would make the collection of each of the η GPs localized to a given radial point of the RDF, together they are denoted $\mathcal{LGP}$.

We assume that the kernel and mean for each localized GP is shared across r values, this implies the GP prior distribution is the same for all points r. $K(X, X)$ is then an N row square matrix rather than the $N \cdot \eta$ for standard GPs. We choose the kernel to be the squared exponential gaussian kernel to assert the RDF is continuous and differentiable for any choice of r_i w.r.t. ϕ [8]. The prior mean of the ith GP is chosen as the first element of an expansion of the RDF in powers of the concentration. This corresponds to the dilute limit and is shown below.

$$K_{mn} = w^2 \exp\left(- \sum_{i=1}^{\dim(\boldsymbol{\theta})} \frac{(\phi_m^{(i)} - \phi_n^{(i)})^2}{2\ell^2_{\phi^{(i)}}} \right), \quad \mu_i(\phi) = \exp(-U_{LJ}(r_i, \sigma, \epsilon)/kT). \tag{4}$$

where U_{LJ} is the Lennard-Jones potential evaluated at r_i and kT is a characteristic energy [7]. It is understood that the σ and ϵ come from components the of ϕ. This choice asserts the LGP will have the correct limiting behavior at low and high r and decreases the error of the model significantly by enforcing physical behavior. Together these define the set of hyperparameters for the LGP as $w, kT, \ell_\epsilon, \ell_\sigma, \ell_{q_H}, \ell_{r_{OM}}, \omega$ where the term ω is the expected noise of the MD simulations prediction of $g_{OO}(r)$.

With the relevant LGP prior defined the expectation of the LGP posterior is then

$$\begin{bmatrix} g_{OO}(r_1;\phi) & \dots & g_{OO}(r_\eta;\phi) \end{bmatrix} \approx \mathbb{E}[\mathcal{LGP}](\phi) = K(\phi, X) \cdot \left(K(X, X) + \omega^2\right)^{-1} \cdot Y \tag{5}$$

which is a η dimensional row vector. The addition of ω^2 is understood to be along the diagonal only. The posterior covariance is then given by

$$K_{Post}(\phi, \phi') = K(\phi, X) \cdot \left(K(X, X) + \omega^2\right)^{-1} K(X, \phi'). \tag{6}$$

Together these define a multivariate normal distribution over $g_{OO}(r_i)$. The set of hyperparameters is then trained by maximizing the the posterior probability of each localized GP using the method

of Sundararajan [17]. Specifically, we perform a leave-one-out N-fold optimization of all 512 training examples. The optimizer used was stochastic gradient descent with momentum [18].

4.2 Markov Chain Monte Carlo

To sample Equation 1, we used Markov chain Monte Carlo (MCMC) [19]. Given a starting state ϕ, to perform MCMC first propose a new state $\phi' \sim q(\phi' \mid \phi)$, then compute the acceptance probability

$$\alpha = \min\left(1, \frac{p(\phi' \mid \mathcal{D})\, q(\phi \mid \phi')}{p(\phi \mid \mathcal{D})\, q(\phi' \mid \phi)}\right). \tag{7}$$

Draw $u \sim \mathcal{U}[0,1]$, and accept ϕ' if $u < \alpha$; otherwise, remain at ϕ.

In the limit of many samples this algorithm approaches samples from the posterior distribution $p(\phi \mid \mathcal{D})$. We used a specialized form of MCMC called Hamiltonian Monte Carlo (HMC) in order to maximize the efficiency of the sampler [9]. This method takes advantage of the volume preserving nature of Hamilton's equations to better sample the state space of $\phi \in \mathcal{A}$ while allowing for highly non-local moves unavailable to typical diffusive sampling techniques.

To apply this algorithm we expanded our state space $\mathcal{A}$ with a conjugate and latent momentum with multivariate normal distribution $\varphi \sim \mathcal{N}(0, M)$.[3] Note that $\dim(\varphi) = \dim(\phi)$. This then gives the Hamiltonian and the equations of motion as

$$H(\phi, \varphi) = \frac{\varphi^T M^{-1} \varphi}{2} + V(\phi), \quad \frac{d\phi^{(i)}}{dt} = \frac{\partial H}{\partial \varphi^{(i)}}, \quad \frac{d\varphi^{(i)}}{dt} = -\frac{\partial H}{\partial \phi^{(i)}}. \tag{8}$$

where M is a hyperparameter mass matrix with size $(\dim(\varphi) \times \dim(\varphi))$. This also makes the assumption that the potential satisfies $V(\phi) = -\ln p(\phi|\mathcal{D})$.

The proposal move in step 1 of MCMC is then given by evolving Hamilton equations forward in time from an initial ϕ and a random $\varphi \sim \mathcal{N}(0, M)$. We implemented the forward evolution using velocity Verlet [21] in PyTorch. This allows us to take advantage of automatic differentiation in order to evaluate the RHS of equation 8. This step involves backpropagation through $H(\phi, \varphi)$ and then via the chain rule the GP posterior mean in equation 5.

This algorithm samples the canonical distribution of statistical mechanics $\propto \exp(-H(\phi, \varphi)/T)$ with the temperature T equal to 1. This implies the acceptance probability is simply the Boltzmann factor $\alpha = \min\left(1, \exp\left(H(\phi, \varphi) - H(\phi', \varphi')\right)\right)$. The mass matrix was tuned by approximating the covariance of ϕ as suggested in [9]. This acts as a preconditioner to the dynamics similarly to the conjugate gradient method and gives more efficient sampling. The timestep and mass were slowly adjusted during an initial burn in period until 0.75 acceptance rate was approximately reached. This then was followed by a much longer 5000 step burn in and then a 5000 step production period. More details of the HMC simulation parameters can be found in the SI python notebook.

5 Results

The training loss of the LGP is shown in the figure 1. The loss is defined as the negative log leave-one-out posterior probability evaluated on the dataset $\mathcal{D}$. In our implementation, the training procedure completed in approximately 8.5 seconds on an a100 GPU. The trained hyperparameters are $\ell_\epsilon = 0.047281$ kcal/mol, $\ell_\sigma = 0.076487$ Å, $\ell_{q_H} = 0.064693$ e, $\ell_{r_{\mathrm{OM}}} = 0.215722$ Å, $w = 0.799165$, $\omega =$

[3] We also place a prior distribution $p(\alpha)$ over the noise α and create a corresponding conjugate momentum β to perform HMC over this dimension as well. This is commonly done for nuisance parameters we wish to marginalize [20].

6×10^{-6}, $kT = 4.141573$ kcal/mol. As shown in the center panel of figure 1, moderate overfitting is observed, particularly near the initial peak of the RDF. However, the overall error magnitude remains less than 0.08. This level of error is negligible relative to the experimental magnitudes reported by Skinner. Additionally, the LGP model is used strictly for interpolation, with no extrapolation permitted due to the constraints of the bounding box $\mathcal{A}$. Therefore, the MD simulations may be safely replaced by the LGP surrogate.

The posterior distribution over the model parameters and expected deviation α is shown in figure 2. This is a corner plot. Conceptually these correspond to marginal distributions of the full joint probability distribution shown in equation 1. The *maximum a posteriori* (MAP) parameters are shown in the figure above each diagonal marginal as well as their standard deviation. These plots were computing using histograms of samples found with the HMC algorithm described above. The total computation time for the tuning of the mass matrix (shown in the top right of figure 2), tuning the timestep, the burn-in and production period of the Markov chain is roughly 20 minutes on an a100 GPU.

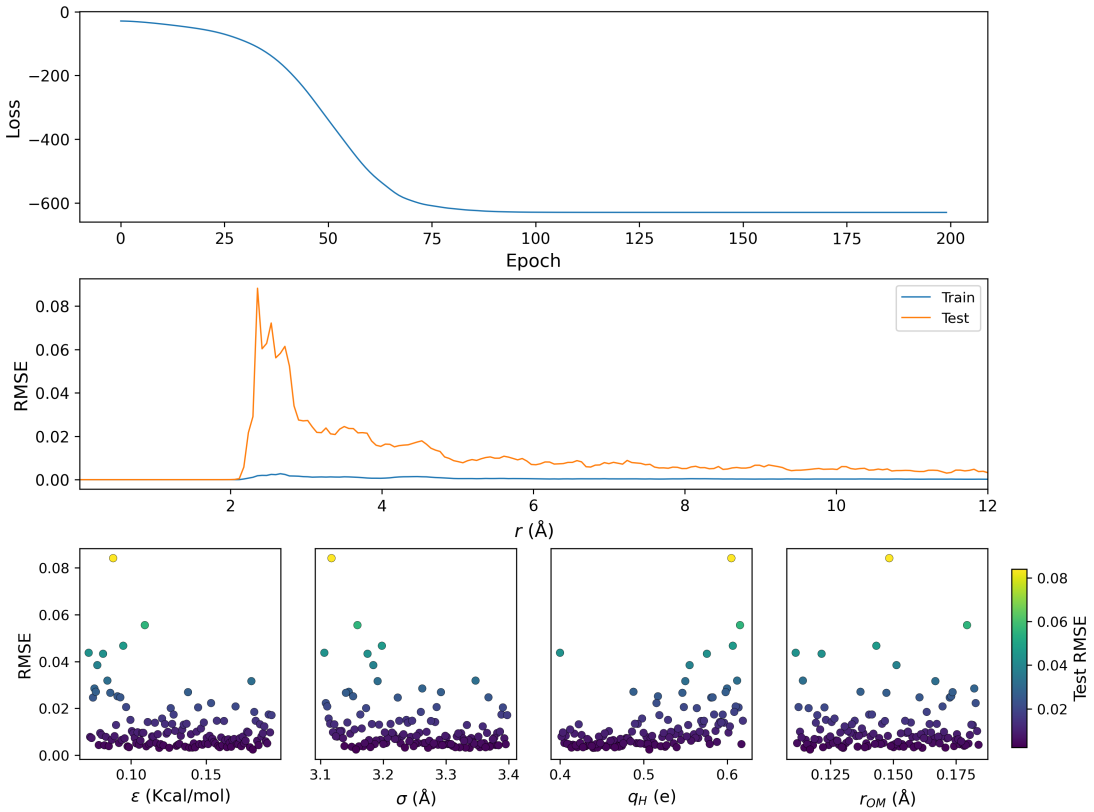

Figure 1: Top: Sum of the negative logarithm of the leave-one-out posterior probability of $\mathcal{D}$ as a function of epoch during hyperparameter training. Middle: RMSE of training vs testing set as a function of the radial component of the RDF. Bottom: Test set RMSE as a function of components of ϕ.

The posterior distribution was then propagated through equation 5 to build up samples of $g_{OO}(r)$. With this set of samples we computed the median as well as upper and lower confidence intervals. These are shown in figure 3. This confidence interval represents the spread of the model as a function of r as induced by the parameter posterior. The MAP parameters were used in LAMMPS. The close agreement between the median and the LAMMPS simulation gives assurance the HMC and LGP are both working. For comparison we computed the difference between all functions shown in the top of

figure 3 and the median predictive. This is shown in the lower panel of figure 3.

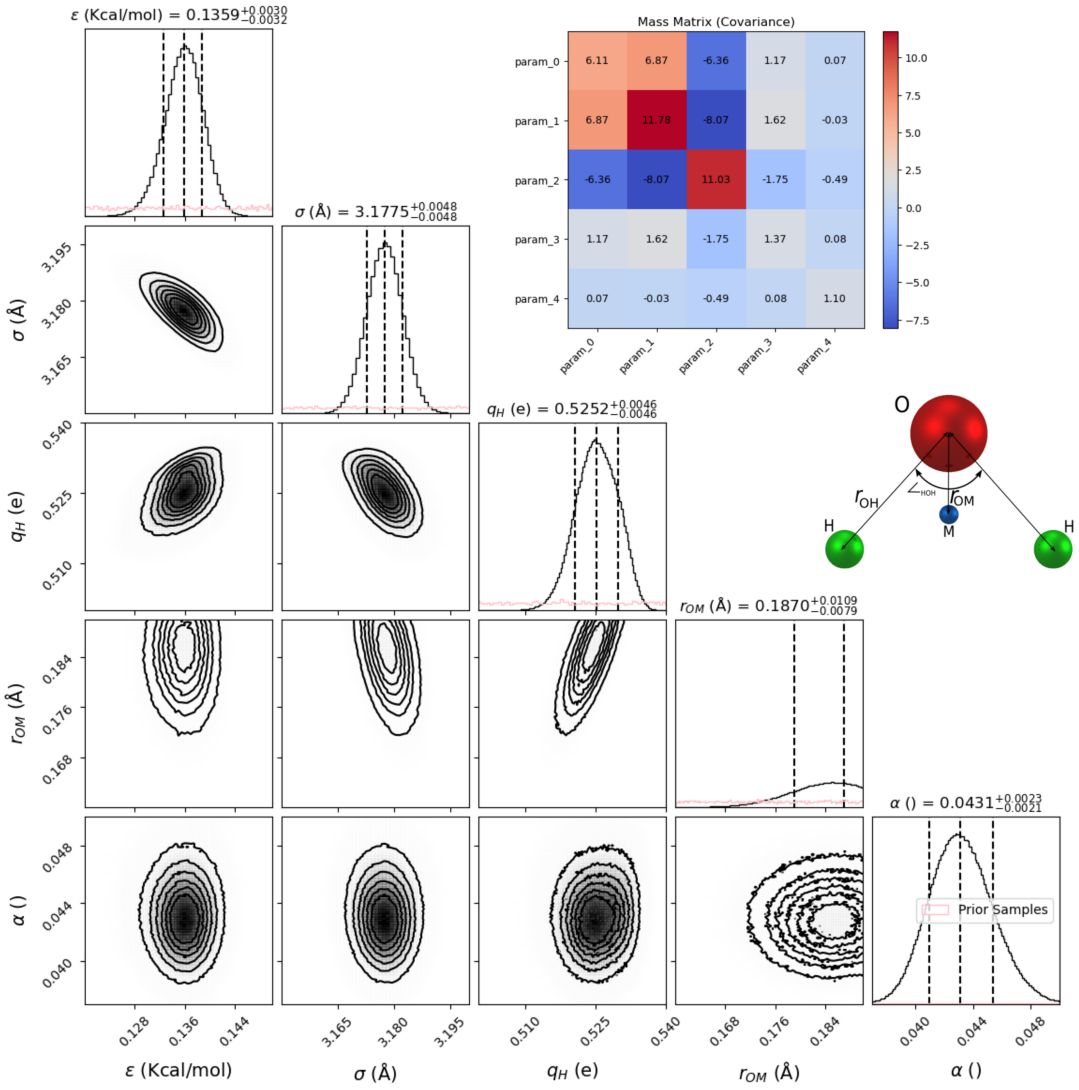

Figure 2: Corner plot showing the joint and marginal posterior histograms of model parameters. In order these are ϵ, σ, q_H, r_{OM}, and α. The 1D histograms on the diagonal are marginalized distributions with 68% credible intervals marked by dashed lines. Off-diagonal contour plots are pairwise histograms between parameters. Insets display the inferred mass matrix used for HMC and a schematic of the molecular geometry. Pink curves = prior. Black curves = posterior.

6 Discussion

Based on the result of our HMC Bayesian optimization in figure 2, the corner plot exhibits unimodal behavior with a well-converged narrow peaks. This shows our model has successfully identified a most probably parameter with high certainty. In other words, our Bayesian optimization has effectively gained information from Skinners scattering data. However, it is also important to note that the posterior distribution on r_{OM} got truncated at the upper bound, implying that the model is trying to distribute probability mass outside of $\mathcal{A}$. A small addition to future work would be extending our sampling range on r_{OM} to prevent this.

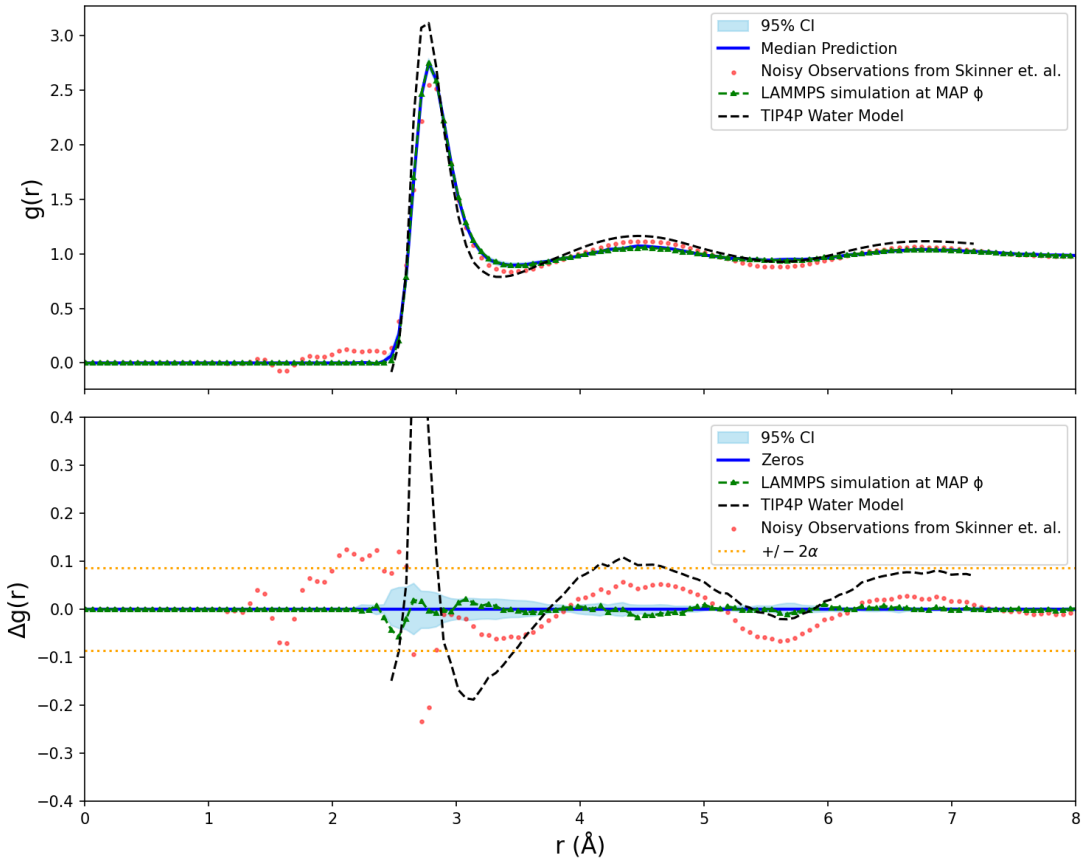

Figure 3: Posterior predictive of the oxygen-oxygen RDF $g_{OO}(r)$. The top panel shows the median prediction (blue) with a 95% credible interval (shaded), compared to experimental observations (red), a simulation at the MAP parameters (green dashed), and the TIP4P model (black dashed). The bottom panel displays the residual $\Delta g_{OO}(r)$ (the difference of the curves and the median). Orange lines represent the expected deviation between the model and data ($\pm 2\alpha$).

The posterior predictive RDF in figure 3 shows great alignment in the first peak. The match in both the intensity and the position as well as broader qualitative agreement is promising. While most observations fall within the expected range shown in the lower panel of figure 3, systematic deviation from the model's credible noiseless interval (blue) suggests underfitting or improper model assumptions. However, in comparison to the TIP4P/2005 radial distribution function, our results show better alignment. This indicates our model had surpassed TIP4P in terms of liquid structure prediction.

The discrepancies past the first peak indicate clear limitations of the rigid water model. In future studies, we should consider using a flexible bond and angle as well as adopting non-coulombic interactions on H. In particular, the dependence of hydrogen-bonding networks on these later peaks implies we are missing important physics of the system. Additionally, the many body behavior, which has been neglected, could result in a better fit. By increasing the complexity of the model we can expect major improvements at the cost of spending more computational resources training an LGP.

Furthermore, we hope to take these results and train additional GPs to predict thermodynamic properties such as the heat capacity, dielectric constant, and self diffusivity. With these GPs we can perform propagation of the posterior in figure 2 to better understand how the incorporation of scattering data affects macroscopic predictions.

References

(1) Shanks, B. L.; Sullivan, H. W.; Hoepfner, M. P. *The Journal of Physical Chemistry Letters* **2024**, *15*, PMID: 39681543, 12608–12618.
(2) Zhang, J.; Pagotto, J.; Duignan, T. T. *J. Mater. Chem. A* **2022**, *10*, 19560–19571.
(3) Skinner, L. B.; Huang, C.; Schlesinger, D.; Pettersson, L. G. M.; Nilsson, A.; Benmore, C. J. *The Journal of Chemical Physics* **2013**, *138*, 074506.
(4) Louis, A. A. *J. Phys. Condens. Matter* **2002**, *14*, 9187.
(5) Abascal, J. L. F.; Vega, C. *The Journal of Chemical Physics* **2005**, *123*, 234505.
(6) Kirkwood, J. G.; Buff, F. P. *J. Chem. Phys.* **1951**, *19*, 774–777.
(7) Hansen, J.; McDonald, I. R., *Theory of Simple Liquids: With Applications to Soft Matter*; Academic Press: 2013.
(8) Rasmussen, C. E.; Williams, C. K. I., *Gaussian processes for machine learning*; MIT Press: Cambridge, Mass, 2006.
(9) Brooks, S.; Gelman, A.; Jones, G.; Meng, X.-L., *Handbook of Markov Chain Monte Carlo*; Chapman and Hall/CRC: 2011.
(10) Garnett, R., *Bayesian Optimization*; Cambridge University Press: 2023.
(11) Allen, M. P.; Tildesley, D. J., *Computer Simulation of Liquids*; Oxford University Press: 2017.
(12) Thompson, A. P.; Aktulga, H. M.; Berger, R.; Bolintineanu, D. S.; Brown, W. M.; Crozier, P. S.; in 't Veld, P. J.; Kohlmeyer, A.; Moore, S. G.; Nguyen, T. D.; Shan, R.; Stevens, M. J.; Tranchida, J.; Trott, C.; Plimpton, S. J. *Comp. Phys. Comm.* **2022**, *271*, 108171.
(13) Sobol', I. *USSR Computational Mathematics and Mathematical Physics* **1967**, *7*, 86–112.
(14) Michaud-Agrawal, N.; Denning, E. J.; Woolf, T. B.; Beckstein, O. *Journal of Computational Chemistry* **2011**, *32*, 2319–2327.
(15) Bishop, C. M., *Pattern Recognition and Machine Learning*; Information science and statistics; Springer: 2006.
(16) Shanks, B. L.; Sullivan, H. W.; Shazed, A. R.; Hoepfner, M. P. *Journal of Chemical Theory and Computation* **2024**, *20*, arXiv:2310.19108 [cond-mat, physics:physics], 3798–3808.
(17) Sundararajan, S.; Keerthi, S. S. *Neural. Comput.* **2001**, *13*, 1103–1118.
(18) Sutskever, I.; Martens, J.; Dahl, G.; Hinton, G. In *Proceedings of the 30th International Conference on Machine Learning*, ed. by Dasgupta, S.; McAllester, D., PMLR: Atlanta, Georgia, USA, 2013; Vol. 28, pp 1139–1147.
(19) Hastings, W. K. *Biometrika* **1970**, *57*, 97–109.
(20) Gelman, A.; Carlin, J. B.; Stern, H. S.; Rubin, D. B., *Bayesian Data Analysis*; Chapman and Hall/CRC: New York, 1995.
(21) Verlet, L. *Phys. Rev.* **1967**, *159*, 98–103.

7 Supplementary Information

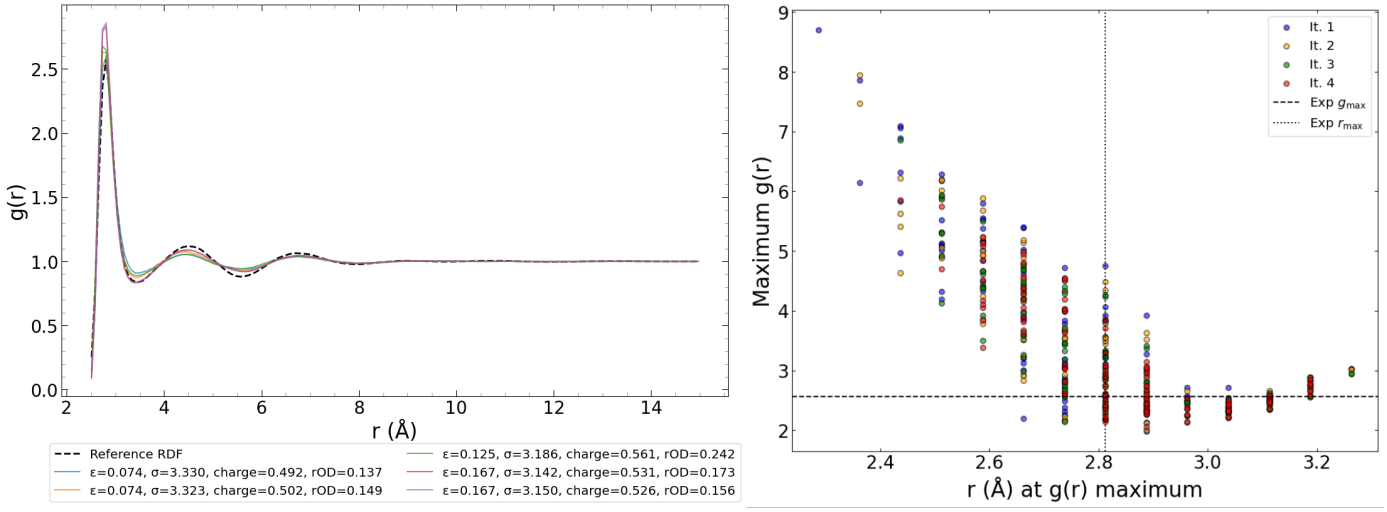

Figure 4: Left: The top 5 lowest sum of squared error simulations. Right: The location and intensity of the first peak in $g_{OO}(r)$ for each training data point. For comparison a reference metric from Skinner's experimental scattering result is shown.

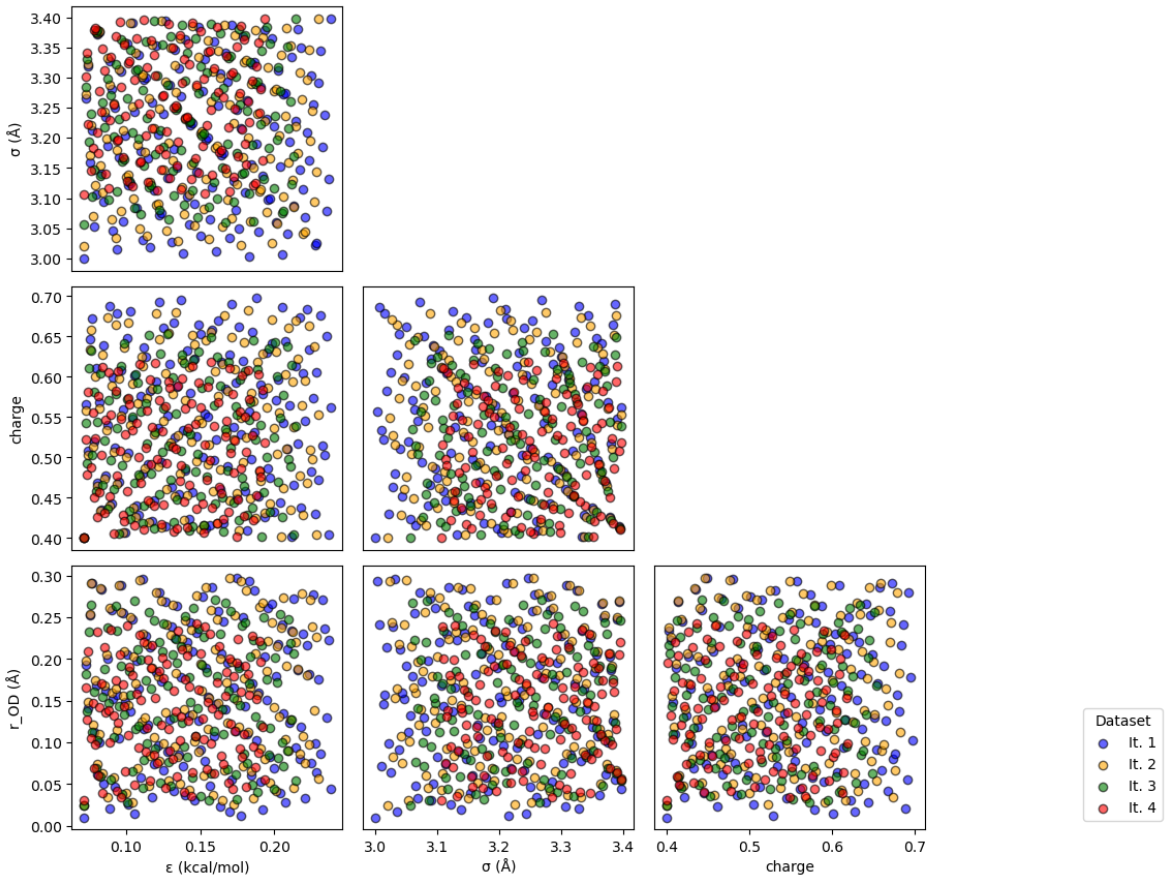

Figure 5: A corner plot of each ϕ in the parameter space $\mathcal{A}$. Each dot corresponds to one MD simulation.