Data-Driven Density Functional Theory: A Case for Physics-Informed Learning

Peter Yatsyshin[1], Serafim Kalliadasis[2] and Andrew Duncan[1,2]
[1]The Alan Turing Institute, London, UK; [2]Imperial College London, London, UK
pyatsyshin@turing.ac.uk

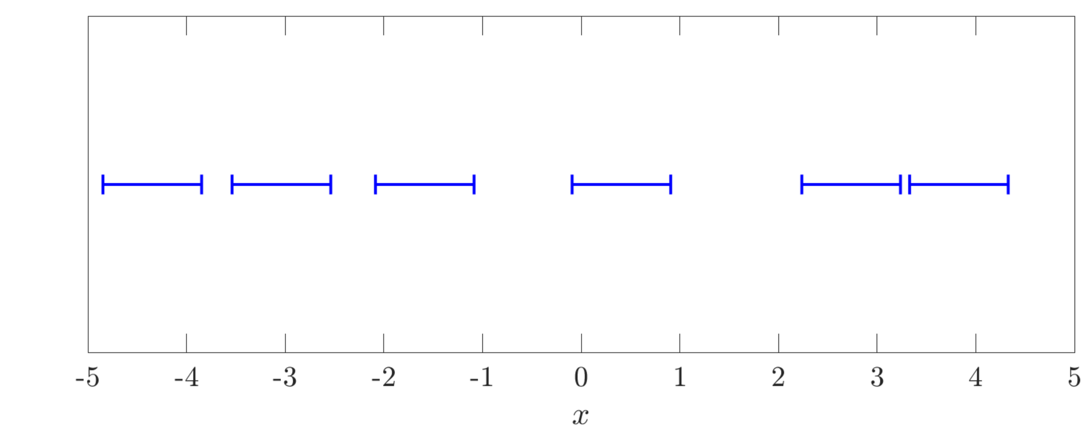
Motivation

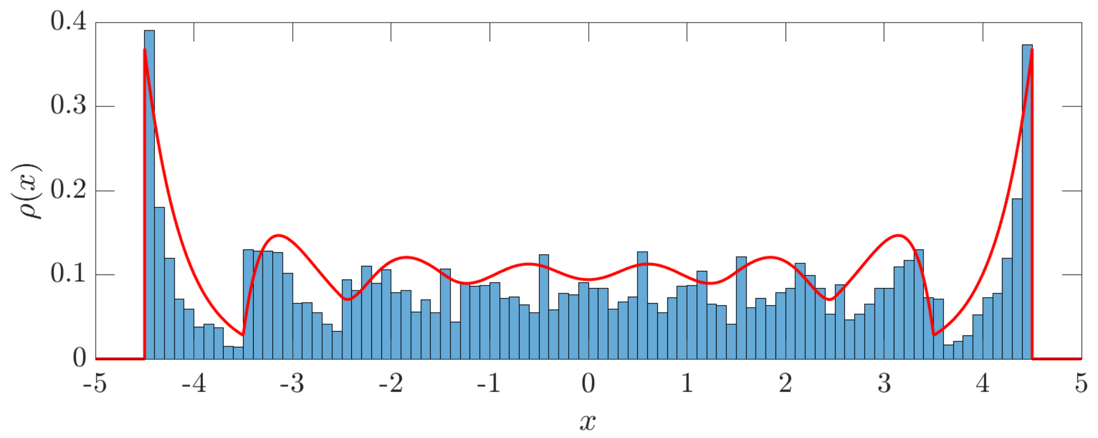
We solve the inverse statistical-mechanical problem: given particle data, reconstruct the system's free energy. We propose a machine-learning algorithm using classical Density Functional Theory (DFT).

What is classical DFT

It is a way to do the statistical mechanics of classical (**NOT quantum**) many-body systems. In DFT the Helmholtz free energy is expressed as a functional $F[\rho]$ of the probability density $\rho(\mathbf{r})$ to find a particle at position $\mathbf{r}$. If $F[\rho]$ is known, we can find $\rho(\mathbf{r})$ by minimising it. After $\rho(\mathbf{r})$ is obtained, it is easy to compute measurable "macroscopic" properties (e.g., pressure, magnetisation, charge distribution, etc.). Classical DFT is used in soft-matter, biology, nanofluidics, chemical engineering, etc.

Our classical many-body system: 1D fluid of hard rods





We generate training datasets by Monte-Carlo

Canonical ensemble:

For N rods, draw $N+1$ gaps, so that the sum of all gaps and rod lengths is L

$$\mathscr{D}_N = \{\ldots(y_1,\ldots y_N)_i,\ldots\} \quad i=1\ldots M$$

Grand-Canonical ensemble:
(With chemical potential μ)

Draw N_i by attempting to insert/delete a particle with μ-dependent probability:

$$P_{\mathrm{del}} = \frac{N_i\exp(-\mu)}{L} \text{ if } N_i > 1 \text{ and } 0 \text{ otherwise}$$

$$P_{\mathrm{ins}} = \frac{L\exp(\mu)}{N_i+1} \text{ if } N_i+1 < L/2R \text{ and } 0 \text{ otherwise}$$

$$N_i \to \langle N_\mu\rangle$$

$$\mathscr{D}_\mu = \left(\mu, \{y_i\}_{i=1}^M, \langle N_\mu\rangle\right)$$

Classical DFT primer

DFT is just a "dialect" of statistical mechanics

$$f_N\left(q^N,p^N,[\phi(\mathbf{r})]\right) = \frac{1}{\Xi}\frac{1}{N!h^{DN}}\exp\left(-\beta(H_N-\mu N)\right)$$

$$H_N = \sum_{i=1}^N \frac{p_i^2}{2m_i} + V(q_1,\ldots q_N) + \int \hat\rho(\mathbf{r})\varphi(\mathbf{r})$$

Mermin's theorem: $\varphi(\mathbf{r}) \leftrightarrow \rho(\mathbf{r})$

Insert one-body density: $\rho(\mathbf{r}) = \sum_{N=0}^\infty \int f_N\left(q^N,p^N,[\phi]\right)\hat\rho(\mathbf{r})dq^Ndp^N$

$$\hat\rho(\mathbf{r}) = \sum_{i=1}^N \delta(\mathbf{r}-q_i)$$

Write down the partition-function and the free energy:

$$\Xi = \sum_{N=0}^\infty \frac{1}{N!h^{DN}}\int \exp\left(-\beta\left(H_N-\mu N\right)\right)dp^Ndq^N \equiv \exp\left(-\beta\Omega[\varphi]\right)$$

Observe: $\rho(\mathbf{r}) = \dfrac{\delta\Omega[\varphi]}{\delta(\varphi(\mathbf{r})-\mu)}$ **Thus,** $\Omega[\rho] = F[\rho] + \int \rho(\mathbf{r})(\varphi(\mathbf{r})-\mu)d\mathbf{r}$

Physics-informed component

$$\Omega[\rho] = \beta^{-1}\int\rho(\mathbf{r})(\ln\lambda^D\rho(\mathbf{r})-1)d\mathbf{r} + F_{\mathrm{ex}}[\rho] + \int\rho(\mathbf{r})(\varphi(\mathbf{r})-\mu)d\mathbf{r}$$

Low-density expansion: $\beta F_{\mathrm{ex}}[\rho] = -\frac{1}{2}\int d\mathbf{r}_1\rho(\mathbf{r}_1)\int d\mathbf{r}_2\rho(\mathbf{r}_2)f(r_{12})$

$$+\frac{1}{6}\int d\mathbf{r}_1\rho(\mathbf{r}_1)\int d\mathbf{r}_2\rho(\mathbf{r}_2)\int d\mathbf{r}_3\rho(\mathbf{r}_3)f(r_{12})f(r_{23})f(r_{31})+\ldots$$

$f(r_{12}) = -1$ Inside overlap

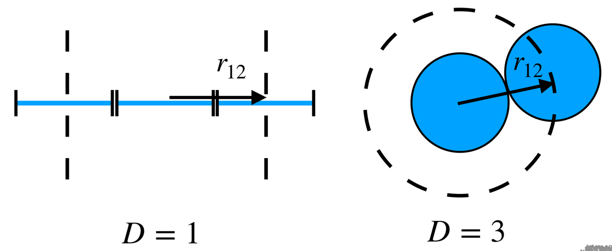
$f(r_{12}) = 0$ Outside overlap

$$f(r_{12}) = -\Theta(\sigma - r_{12})$$

$$n_i = \omega_i \otimes \rho$$

$$= \int d\mathbf{r}\left(-\sum_{ij} n_i(\mathbf{r})n_j(\mathbf{r})\right) + \ldots$$

$$\approx \int d\mathbf{r}\Phi(n_0(\mathbf{r}), n_1(\mathbf{r}), \ldots)$$



$f = -2\omega_0 \otimes \omega_1$

$\omega_0(x) = \frac{1}{2}\delta\left(\frac{\sigma}{2}-|x|\right)$

$\omega_1(x) = \Theta\left(\frac{\sigma}{2}-|x|\right)$

$f = \sum_{ij}\omega_i\otimes\omega_j$

Let's learn the distributional model

$$F_{\mathrm{ex}}[\rho] = \beta^{-1}\int d\mathbf{r}\Phi\left(\{n_i(\mathbf{r})\}\right)$$

$$\eta(x) = \int_{x-\sigma/2}^{x+\sigma/2}\rho(x)dx = \omega_1\otimes\rho$$

$$n_0(x) = \frac{\rho(x-\sigma/2)+\rho(x+\sigma/2)}{2} = \omega_0\otimes\rho$$

Inference model: polynomial parametrisation of $\Phi(n_0,\eta) = \Phi(n_0,\eta\mid Q)$

$$\Phi(n_0,\eta\mid Q) = \left(a_{N_1}n_0(x)^{N_1} + a_{N_1-1}n_0(x)^{N_1-1} + e^{a_0}\right)\left(b_{N_2}\eta(x)^{N_2} + \ldots b_0\right),$$

$$Q = \left(a_{N_1},\ldots a_0, b_{N_2},\ldots b_0\right)^T$$

Bayesian inference of model parameters Q

Likelihood: $P(\mathscr{D}_\mu\mid Q) = \prod_{i=1}^M \rho(y_i\mid Q)$, $\mathscr{D}_\mu = \left(\mu, \{y_i\}_{i=1}^M, \langle N_\mu\rangle\right)$

Gaussian prior: $\mathscr{N}(Q\mid 0, \Sigma_Q)$

Posterior: $P(Q\mid\mathscr{D}_\mu) \propto \mathscr{N}(Q\mid\bar Q,\Sigma_Q)\prod_{i=1}^M\rho(y_i\mid Q)$

Learning algorithm

I. Obtain $\rho(x)$ for the current Q from the DFT Euler-Lagrange (EL)

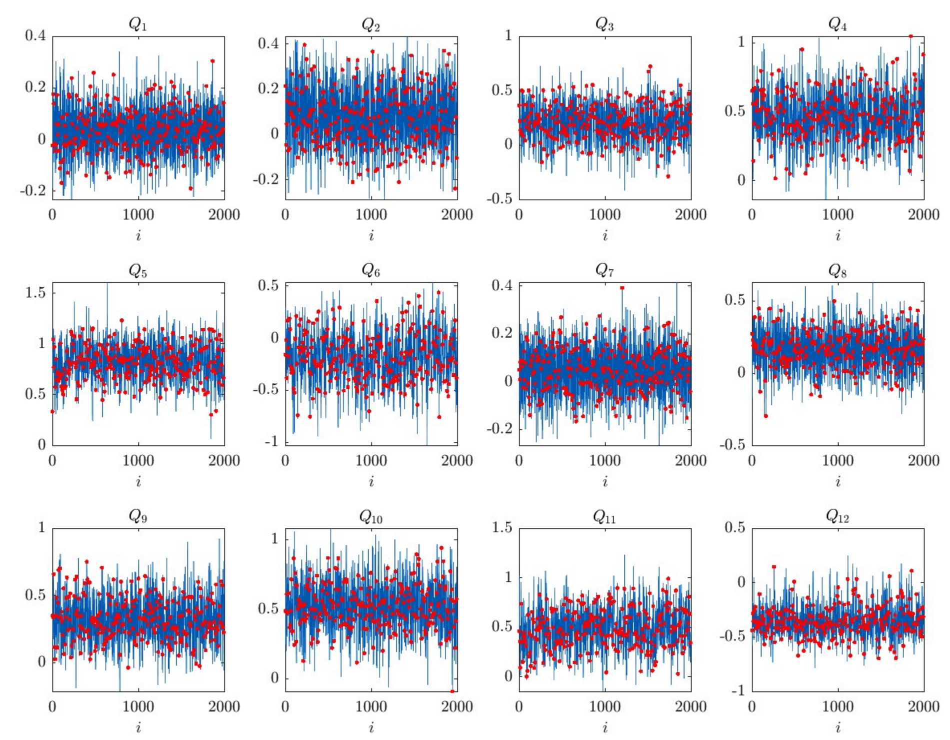
$$T\ln\rho(x) + T\left(\omega_v\circledast\frac{\partial\Phi(n_0,\eta\mid Q)}{\partial\eta(x)} + \omega_s\circledast\frac{\partial\Phi(n_0,\eta\mid Q)}{\partial n_0(x)}\right) - \mu = 0, \int_{-L/2}^{L/2}\rho(x)dx = \langle N_\mu\rangle$$

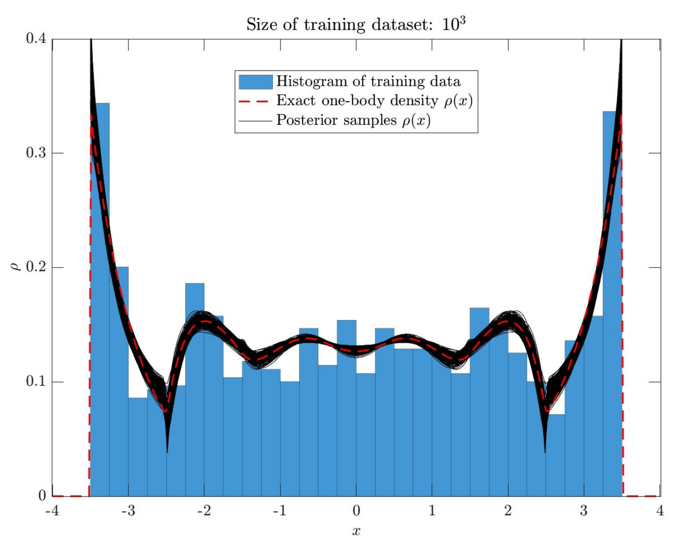
II. Compute $\nabla_Q\rho_Q(x)$ via "adjoint method". Denote EL equation: $g\left(\rho_Q(x), Q\right) = 0$

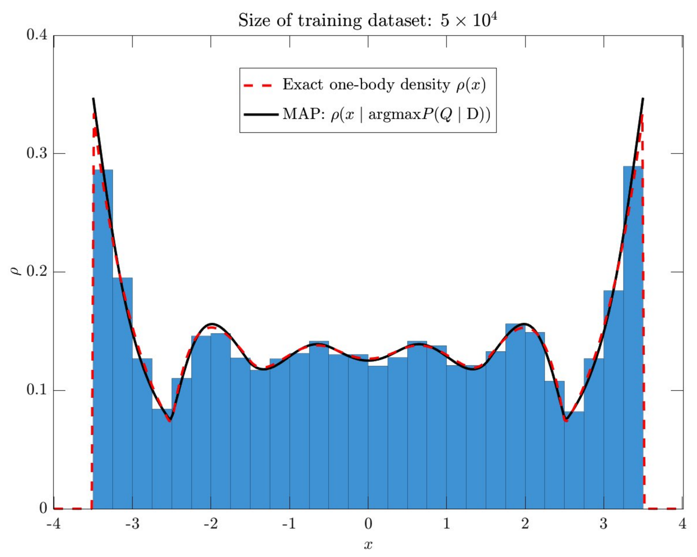
$$\underbrace{d_Q g(\rho_Q, Q)}_{n_Q\times 1} = 0 \qquad \underbrace{\frac{\partial g}{\partial\rho}}_{n_\rho\times n_\rho}\overbrace{\nabla_Q\rho_Q}^{n_\rho\times n_Q} = -\underbrace{\frac{\partial g}{\partial Q}}_{n_\rho\times n_Q} \quad \text{linear system for } \nabla_Q\rho_Q(x)$$

III. Compute $\nabla_Q P(Q\mid\mathscr{D}_\mu)$ and perform HMC jump in Q

Typical HMC trace plots







Gaussian random field for chemical potential μ

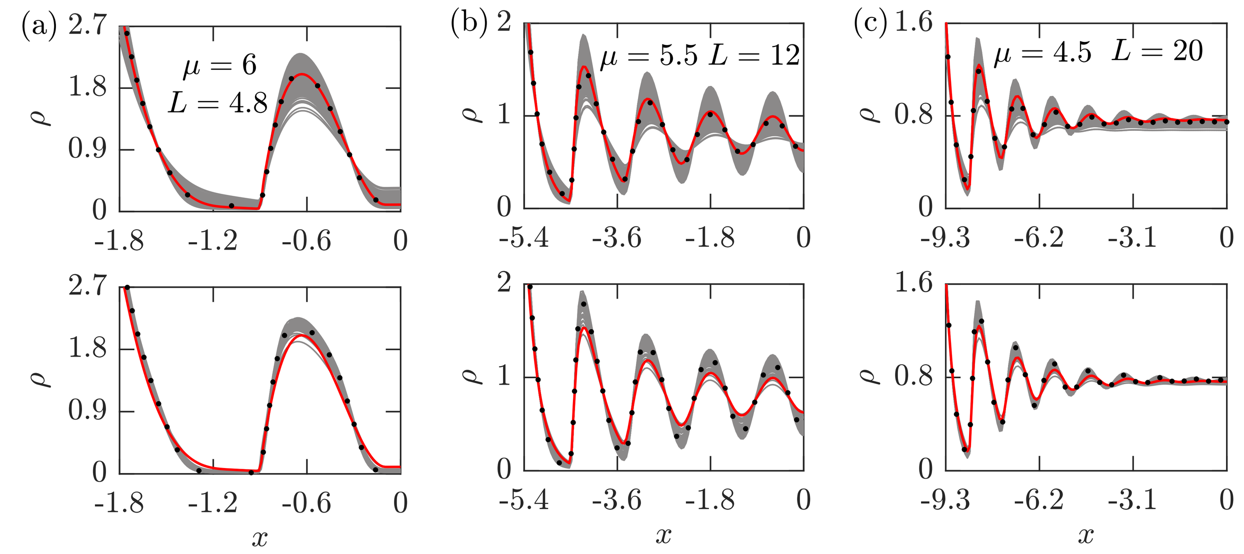
$$\Phi(n_0,\eta\mid\mu,\alpha) \equiv \Phi(n_0,\eta\mid Q(\mu\mid\alpha)), \quad Q(\mu\mid\alpha) = A\left(\mu^M,\mu^{M-1}\ldots 1\right)^T,$$

where A is an $N_Q\times(M+1)$ matrix, and α is the flattened A

Training dataset: $\mathscr{D} = \{\mathscr{D}_{\mu_n}\}_{n=1}^K \equiv \left\{\left(\mu_n, \{y_{i|\mu_n}\}_{i=1}^{M_{\mu_n}}, \langle N_{\mu_n}\rangle\right)\right\}_{n=1}^K$

Likelihood: $P(\mathscr{D}\mid\alpha) = \prod_{n=1}^K\prod_{i=1}^{M_{\mu_n}}\rho(y_{i|\mu_n}\mid\mu_n,\alpha)$

Trained on integer μ = -2, -1, 0, 1, 2, 3, 4, 5; $M_{\mu_n} = 10^4$; pore width $L = 8$



Learn canonical functional: Fixed N

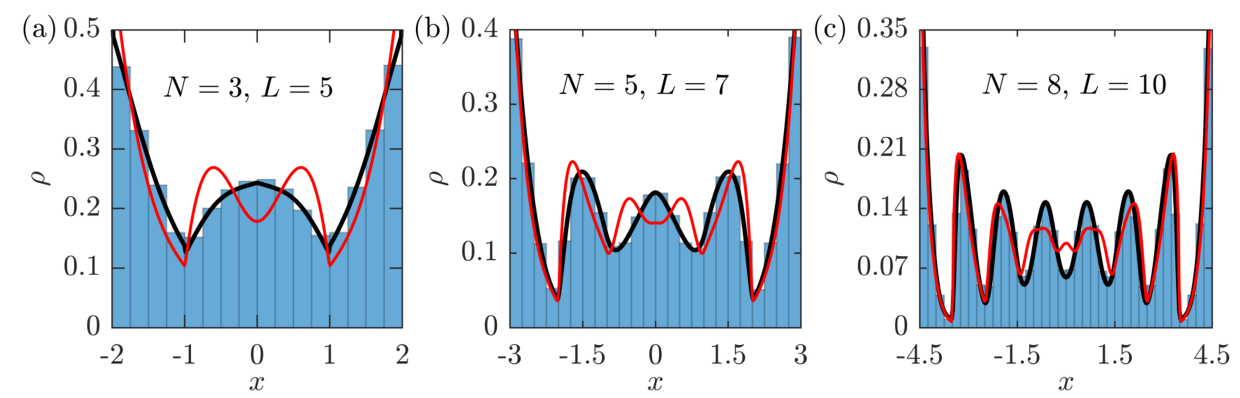
$$\mathscr{L}[\rho] = F_N[\rho] + \int\rho(\mathbf{r})\left(V(\mathbf{r})-\lambda\right)d\mathbf{r}.$$

$\mathscr{L}$ is simply a Lagrangian (it does not give the partition function)

λ is not a thermodynamic field. Only a Lagrange multiplier

Training dataset: $\mathscr{D}_N = \{L_n, \{(y_1,\ldots y_N)_i\}_{i=1}^{M_n}\}_{n=1}^K$.

Trained on K = 6 equispaced $L_n \in [L-2, \quad L+2]$; $M_n = 10^4$ coordinates



Data efficiency: compare with a black-box model

$$\rho(x\mid\mu) = \sum_{i=1}^{N_f}\alpha_i(\mu)\exp\left(-(x-p_i)^2/w_i^2(\mu)\right), \sum_j^{N_f}\alpha_j(\mu) = 1$$