Lecture 9

Learning a Potential Energy Surface

Contents

1 **Potential Energy Surface** 1

2 **Feature Engineering** 3

3 **Model Selection** 9

4 **Training Data** 11

5 **Lab** 12

In this lecture, we focus on one of the main applications of AI/ML in chemistry: learning a potential energy surface (PES). Unlike previous lectures, this one is organized around the problem itself, followed by the construction of the necessary components to solve it using a data-driven approach.

A comprehensive tutorial by Tokita and Behler [1] overlaps with some topics in this lecture and provides detailed guidance on training feed-forward neural networks with atomic-centered symmetry functions (ACSF). It also discusses active learning, which will be covered in a future lecture.

In addition, we introduce another feature engineering method, the Smooth Overlap of Atomic Positions (SOAP), and review Gaussian Process Regression (GPR) as an alternative approach for learning PES.

1 Potential Energy Surface

A potential energy surface (PES) is a mathematical map that describes the energy of a chemical system as a function of its geometry. The chemical system may be a single molecule, a collection of molecules, a polymer, a solid-state material, or a mixture of different phases. The geometry of the system is defined by the positions of all atoms.

Mathematically, a PES is a multivariable function

$$E(\mathbf{R}_1, \mathbf{R}_2, \cdots, \mathbf{R}_N) = E(\{\mathbf{R}_i\}_{i=1}^N), \tag{1}$$

where $\mathbf{R}_i$ denotes the three-dimensional Cartesian coordinate of the i-th atom, and N is the total number of atoms in the system.

Mathematically important quantities of the PES include global and local minima, first-order saddle points[1], as well as first derivatives (gradients) and second derivatives (Hessians) with respect to the atomic positions $\{\mathbf{R}_i\}$.

An example of a PES with 2 coordinates is shown in Fig. 1.

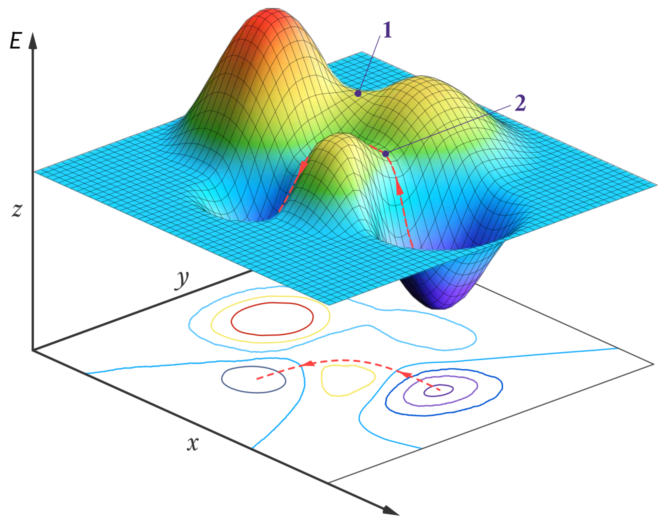

Figure 1: Potential energy surface (PES) with two coordinates. Points 1 and 2 indicate first-order saddle points, the deep blue valleys correspond to two local minima, and the red line shows a reaction pathway. Both the PES landscape and contour plot are displayed. (CC BY-NC; Ümit Kaya via LibreTexts)

Applications

A wide range of chemically relevant information can be extracted from the PES and its derivatives. Applications include:

- **Geometry optimization and conformer sampling**
  - The global minimum of the PES corresponds to the most stable equilibrium structure.
  - Local minima correspond to stable conformers, which can be used for thermodynamic sampling and free energy evaluation.
- **Chemical reaction pathways**

[1] A 1st-order saddle point is a maximum along one coordinate (reaction coordinate) and a minimum along all other coordinates. Mathematically, the gradient = 0 for all coordinates, and the Hessian matrix has exactly one negative eigenvalue.

  - Reactants, products and intermediates correspond to local minima on the PES.
  - First-order saddle points correspond to transition states.
  - Connecting reactants, transition states, and products identifies the reaction pathways.
- **Molecular dynamics (MD)**
  - MD generates time-resolved trajectories, allowing computation of dynamical and thermodynamic properties by sampling motion on the PES.
  - MD simulations require atomic positions $\{\mathbf{R}_i\}$ and forces on atoms $\{\mathbf{F}_i\}$, where $\mathbf{F}_i = -\nabla_{\mathbf{R}_i} E(\{\mathbf{R}_j\})$.
- **Vibrational spectroscopy**
  - Requires evaluation of the second derivatives (Hessian) of the PES, which is computationally expensive.
  - Provides vibrational modes of atoms in molecules or phonons in crystals, directly related to IR and Raman spectra.
  - The vibrational frequencies determine the zero-point energy (ZPE) of the system.

All derivatives are evaluated under the Born-Oppenheimer approximation, assuming that electrons relax much faster than nuclei. Therefore, the electronic PES remains valid for all the above tasks.

A summary of the relationships between PES mathematical quantities, their chemical meanings, and applications is provided in Table 1.

Table 1: Chemical interpretations and applications of PES quantities.

Math Quantity	**Chemical Meaning**	**Applications**
Global minimum	Most stable structure	Equilibrium geometry
Local minima	Isomers/conformers, intermediates	Conformer sampling, reaction
Saddle point	Transition state	Reaction kinetics / path-finding
First derivative	Atomic forces	Molecular dynamics
Second derivative	Vibrational modes and frequencies	IR/Raman spectroscopy, ZPE

2 Feature Engineering

Since the potential energy surface (PES) depends on the positions of all atoms in the system $\{\mathbf{R}_i\}$, the most direct representation would use the atomic types together with their Cartesian coordinates. However, the total energy of a molecular system is **invariant under global translation and rotation**.

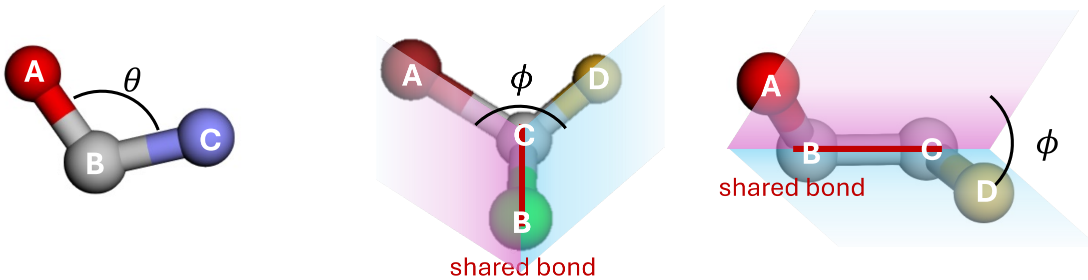

Figure 2: Bond angle θ and dihedral angle ϕ.

Therefore, any physically meaningful representation must respect these invariances.

This motivates the construction of feature representations that are explicitly **rotationally and translationally invariant**.

2.1 Internal coordinates

In the previous lecture, we introduced the **distance-based** representation $\{d_{ij}\}$, which satisfies translational and rotational invariance. However, pairwise distances alone do not always uniquely determine a three-dimensional structure, a limitation known as the *reconstruction problem*. Furthermore, in molecular systems, the energy cost associated with bond stretching is physically distinct from that of bond bending. Without explicit angular information, a model cannot easily distinguish these modes of deformation, and therefore struggles to represent restoring forces associated with bending vibrations.

To resolve this, we augment distance information with **bond angles** and **dihedral angles**.

The combined set of distances, angles, and dihedrals is referred to as **internal coordinates**.

2.1.1 Bond angle and dihedral

- **Bond angle:** defined as the angle between two bonds A–B and B–C that share a common atom B. A bond angle involves three atoms and is denoted as

$$\theta = \angle ABC.$$

- **Dihedral (torsion) angle:** defined as the angle between the two planes formed by atoms A–B–C and B–C–D, which share the central bond B–C. A dihedral angle involves four atoms and is denoted as

$$\phi = \text{dihedral}(A-B-C-D).$$

  Sign of the dihedral. The dihedral angle takes values in the range $(-180°, 180°]$. Its sign is defined by a right-hand convention, illustrated in the example box below.

Stereochemistry. Signed dihedral angles are essential for distinguishing stereochemical arrangements, including chirality (R/S), anti/gauche conformations in alkanes, E/Z isomerism in alkenes, and cis/trans conformations in rings.

Bond angles and dihedral angles are illustrated in Fig. 2.

Example: Sign of the dihedral angle

To determine the sign of dihedral$(A\text{–}B\text{–}C\text{–}D)$, consider the following convention:

Look down the B–C bond, with atom B closer to the observer:

- Clockwise rotation from A toward D → **negative** dihedral.
- Counter-clockwise rotation from A toward D → **positive** dihedral.

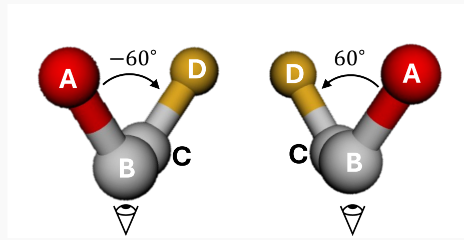

Figure 3: Sign convention for dihedral angles.

In principle, one could enumerate all bond lengths, bond angles, and dihedral angles in a molecule. However, the number of such combinations grows combinatorially with system size, making this approach impractical and poorly scalable.

In the following sections, we introduce two practical strategies for encoding internal coordinates in a way that is both physically meaningful and scalable for ML models.

2.2 Atom-Centered Symmetry Functions (ACSF)

Atom-Centered Symmetry Functions (ACSF), also called Behler–Parrinello Symmetry Functions (BPSF), are constructed based on the following idea:

If atoms are very far apart, the total energy of the system can be expressed as a sum of atomic contributions:

$$E_{\text{tot}} = \sum_i E_i. \tag{2}$$

However, bonded atoms interact with each other, so the total energy can be expanded in many-body terms:

$$E_{\text{tot}} = \sum_i E_i + \sum_{ij} E_{ij} + \sum_{ijk} E_{ijk} + \cdots, \tag{3}$$

where $E_i, E_{ij}, E_{ijk}, \ldots$ are the 1-body, 2-body, 3-body, . . . contributions. These many-body terms can be absorbed into an *effective atomic energy* $\tilde{E}_i$ such that

$$E_{\text{tot}} = \sum_i \tilde{E}_i. \tag{4}$$

For each atom, only nearby atoms contribute significantly to $\tilde{E}_i$, so it is sufficient to consider the *local environment* of the atom.

Behler and Parrinello introduced a set of symmetry functions $\{G_i\}$ to represent each atomic environment, which are then used as input to a neural network.

2.2.1 Symmetry Functions

Behler–Parrinello symmetry functions are designed to satisfy: (1) translational invariance, (2) rotational invariance, (3) permutation invariance of identical atoms, and (4) locality (only neighbors within a cutoff radius contribute). These properties ensure that the learned potential energy surface respects the fundamental symmetries of the underlying physics.

We first define a cutoff function to determine which atoms are included in the *local environment* of atom i:

$$f_c(R_{ij}) = \begin{cases} \frac{1}{2}\left[\cos\left(\frac{\pi R_{ij}}{R_c}\right) + 1\right], & R_{ij} \le R_c \\ 0, & R_{ij} > R_c \end{cases} \tag{5}$$

where R_c is the cutoff radius. Only neighbors j within R_c contribute to the symmetry functions.

The symmetry functions include:

- **Radial symmetry functions (G2, two-body).** These encode distance information around atom i:

$$G_{2,i} = \sum_{j \neq i} f_c(R_{ij})\, e^{-\eta(R_{ij}-R_s)^2}, \tag{6}$$

  where η is a width parameter and R_s is the Gaussian center. Multiple (η, R_s) pairs are used to sample the local radial density.

- **Angular symmetry functions (G4, three-body).** These encode bond-angle correlations:

$$G_{4,i} = 2^{1-\zeta} \sum_{j,k \neq i} \left(1 + \lambda \cos\theta_{ijk}\right)^{\zeta} e^{-\eta(R_{ij}^2+R_{ik}^2+R_{jk}^2)} f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}), \tag{7}$$

  where θ_{ijk} is the bond angle at atom i, $\lambda = \pm 1$ distinguishes acute vs. obtuse angles, ζ controls angular resolution, and η controls radial decay. Multiple (ζ, λ, η) combinations are typically used.

The higher body terms are implicitly included in the G_4 terms, approximated by the linear combinations of 2-body and 3-body terms.

In practice, **species-resolved** G2 and G4 functions are used:

- Each G_{2i} is a vector of length N_{species} (one component per neighbor species).
- Each $G4_i$ is a vector of length $N_{\text{species-pairs}} = \frac{N_{\text{species}}(N_{\text{species}}+1)}{2}$

We also include the **G1 term**, which is the sum of $f_c(R_{ij})$ over neighbors of each species.

Thus, the total feature vector for *each atom* has length:

$$N_{\text{features}} = N_{\text{species}} + N_{\text{G2}} \times N_{\text{species}} + N_{\text{G4}} \times \frac{N_{\text{species}}(N_{\text{species}} + 1)}{2}. \tag{8}$$

For a system (e.g., a molecule), the feature matrix is of size

$$(N_{\text{atoms}}, N_{\text{features}}) \tag{9}$$

2.3 Smooth Overlap of Atomic Positions (SOAP)

The Smooth Overlap of Atomic Positions (SOAP) [2] representation is another widely used descriptor for local atomic environments that is explicitly invariant to translations, rotations, and permutations of identical atoms. In contrast to ACSF, which relies on discrete distance- and angle-based functions, SOAP represents the local environment as a *continuous atomic density*.

Instead of treating atoms as point particles, each atom is represented by a Gaussian density. For a position $\mathbf{r}$ in space, the total atomic density is defined as

$$\rho(\mathbf{r}) = \sum_j \exp\left[-\frac{|\mathbf{r} - \mathbf{R}_j|^2}{2\sigma^2}\right], \tag{10}$$

where $\mathbf{R}_j$ denotes the position of atom j, and σ controls the width of the Gaussian smearing.

For a given atom i, we place the origin at $\mathbf{R}_i$ and define $\mathbf{r}_i = \mathbf{r} - \mathbf{R}_i$, the vector from atom i to position $\mathbf{r}$. The local atomic environment around atom i is then described by

$$\rho(\mathbf{r}_i) = \sum_{j \in \text{neighbors}} \exp\left[-\frac{|\mathbf{r}_i - \mathbf{R}_{ij}|^2}{2\sigma^2}\right], \tag{11}$$

where $\mathbf{R}_{ij} = \mathbf{R}_j - \mathbf{R}_i$ is the displacement vector from atom i to atom j. Only neighboring atoms within a cutoff radius R_c are included in the sum.

The local density is expanded in a basis of *radial basis functions* (RBFs) $g_n(r)$ and *spherical harmonics* $Y_{lm}(\hat{\mathbf{r}})$, a procedure known as the **radial–angular decomposition**:

$$\rho(\mathbf{r}_i) \approx \sum_{nlm} c_{nlm}^{(i)} g_n(r_i) Y_{lm}(\theta, \phi), \tag{12}$$

where $c_{nlm}^{(i)}$ are the expansion coefficients, $r_i = |\mathbf{r}_i|$, and (θ, ϕ) are the polar and azimuthal angles of the unit vector $\hat{\mathbf{r}}_i = \mathbf{r}_i / |\mathbf{r}_i|$.

Common choices for the radial basis functions include Gaussian-type orbitals (GTOs) and orthogonal polynomials. The `dscribe` package adopts GTOs by default.

To achieve rotational invariance, the expansion coefficients are combined into the *power spectrum*:

$$p_{nn'l}^{(i)} = \sum_{m} c_{nlm}^{(i)*} c_{n'lm}^{(i)}, \tag{13}$$

where the summation over m removes the dependence on spatial orientation. The vector $\mathbf{p}^{(i)}$ constitutes the SOAP descriptor for atom i.

The length of the SOAP feature vector $\mathbf{p}^{(i)}$ is

$$\frac{(n_{\max} + 1) n_{\max}}{2} \times (l_{\max} + 1) \tag{14}$$

since the angular momentum quantum number l runs from 0 to $l_{\max}$. SOAP descriptors are controlled by several hyperparameters: the cutoff radius R_c, Gaussian width σ, number of radial basis functions $n_{\max}$, and maximum angular momentum $l_{\max}$. In practice, SOAP features are most commonly used with kernel-based methods (e.g., Gaussian Process Regression) or neural networks to learn potential energy surfaces.

2.3.1 Connection to Quantum Mechanics

The construction of SOAP closely mirrors concepts from electronic structure theory:

- **Density representation.** In quantum mechanics, electrons are described by wavefunctions and continuous density distributions rather than point particles. Although nuclei exhibit much weaker quantum effects, representing atomic environments as smooth densities enables stable, differentiable similarity measures, which are particularly well suited for kernel-based methods such as Gaussian Process Regression. This **smearing** technique is widely used to improve numerical stability in both computational chemistry and ML models, for example in variational autoencoders (VAEs).

- **Basis expansion.** In electronic structure theory, wavefunctions are expressed as linear combinations of basis functions, avoiding explicit treatment of the continuous spatial variable $\mathbf{r}$. SOAP adopts the same strategy by encoding the atomic density through expansion coefficients $c_{nlm}^{(i)}$.
- **Hydrogen-like angular structure.** The use of radial functions and spherical harmonics resembles the solutions of the hydrogen atom. Here, n plays a role analogous to the principal quantum number, l corresponds to orbital angular momentum ($s, p, d, f, \ldots$), and m labels spatial orientation (e.g., p_x, p_y, p_z).

3 Model Selection

We consider two models for learning the PES:

1. A feedforward neural network (FNN).
2. Gaussian process regression (GPR).

Both ACSF and SOAP descriptors are compatible with FNN and GPR. SOAP naturally provides a *kernel* for comparing atomic environments. For two environments i and j, the SOAP kernel is defined as

$$\mathcal{K}_{ij}^{\mathrm{SOAP}} = \frac{\mathbf{p}^{(i)} \cdot \mathbf{p}^{(j)}}{\|\mathbf{p}^{(i)} \cdot \mathbf{p}^{(j)}\|}, \tag{15}$$

which quantifies the similarity between the two environments. The kernel can be sharpened using a power, $(\mathcal{K}_{ij}^{\mathrm{SOAP}})^{\zeta}$.

3.1 Feedforward Neural Network (FNN)

We train a neural network to predict the atomic energy E_i from the corresponding feature vector:

$$\text{Feature vector for atom } i \quad \rightarrow \quad \text{NN} \quad \rightarrow \quad E_i,$$

where the feature vector comes from either ACSF or SOAP.

Since chemical systems may contain multiple element types, we use an element-specific NN for each type.

The training dataset $\{\mathbf{R}_i, E_{\mathrm{tot}}, \mathbf{F}_i\}$ ($\mathbf{F}_i$ optional) is usually generated from electronic structure calculations such as density functional theory (DFT). Including atomic forces

$$\mathbf{F}_i = -\frac{\partial E_{\mathrm{tot}}}{\partial \mathbf{R}_i}$$

can improve training and accelerate convergence.

The hyperparameters of the symmetry functions (η, ζ, λ, etc.) are pre-selected. For each atom, the corresponding feature vector is computed and fed into its element-specific neural network.

The network is trained by minimizing the mean squared error (MSE) between the predicted total energy

$$E_{\text{tot}}^{\text{pred}} = \sum_i \tilde{E}_i$$

and the reference energy $E_{\text{tot}}^{\text{ref}}$:

$$\mathcal{L} = \left(E_{\text{tot}}^{\text{pred}} - E_{\text{tot}}^{\text{ref}}\right)^2 .$$

3.2 Gaussian Process Regression (GPR)

Since the PES is highly non-linear, linear regression is generally insufficient. Kernel-based methods such as Gaussian Process Regression (GPR) are more suitable. GPR is a non-parametric, kernel-based regression method that can learn smooth PES from a limited number of molecular configurations. It is particularly effective for small molecules and when descriptors (e.g., SOAP) are high-dimensional and continuous.

3.3 GPR Formalism (Review)

Given a dataset of feature-label pairs $\{(\mathbf{x}_i, y_i)\}$, the predicted label y_* for a new feature $\mathbf{x}_*$ is assumed to follow a Gaussian distribution:

$$p(y_*) \sim \mathcal{N}(\mu_*, \sigma_*^2), \tag{16}$$

where the mean and variance are determined by the known data and their similarity to $\mathbf{x}_*$ measured via a kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$:

$$\mu_* = \mathbf{k}_*^T K^{-1}\mathbf{y}, \quad \sigma_*^2 = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T K^{-1}\mathbf{k}_*, \tag{17}$$

with

- $\mathbf{k}_*$ having entries $k(\mathbf{x}_*, \mathbf{x}_i)$,
- K with entries $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$,
- $\mathbf{y}$ being the vector of known labels.

3.4 Training Process

GPR "parameters" correspond to kernel hyperparameters. For example, the RBF kernel has length scale ℓ and noise level σ. Training involves finding the optimal (ℓ, σ) by maximizing the log marginal likelihood:

$$\log p(\mathbf{y} \mid X) = -\frac{1}{2}\mathbf{y}^\top K^{-1}\mathbf{y} - \frac{1}{2}\log|K| - \frac{n}{2}\log 2\pi, \tag{18}$$

where n is the number of data points and $p(\mathbf{y} \mid X)$ is a multivariate Gaussian:

$$p(\mathbf{y} \mid X) = \frac{1}{\sqrt{(2\pi)^n |K|}} \exp\left[-\frac{1}{2}(\mathbf{y} - \boldsymbol{\mu})^T K^{-1}(\mathbf{y} - \boldsymbol{\mu})\right]. \tag{19}$$

Often, the labels are normalized such that $\mu = 0$, reducing the likelihood to Eq. (18).

Training steps:

1. Compute feature vectors for all training configurations (e.g., SOAP vectors for each atom, summed or flattened per molecule).
2. Construct the kernel matrix K using the chosen kernel function.
3. Optimize kernel hyperparameters by maximizing the log marginal likelihood.
4. Once trained, predict energies for new configurations using μ_*, and optionally obtain uncertainty σ_*.

4 Training Data

To train a potential energy surface (PES), we sample molecular configurations from the PES and evaluate the energy of each configuration using electronic structure methods such as density functional theory (DFT). Since we are usually interested in *chemically relevant configurations*, we generate *thermally accessible configurations* by performing molecular dynamics (MD) simulations at a given temperature.

The typical workflow is:

- Run an MD simulation to produce a time-dependent series of molecular configurations, called the **MD trajectory**. An example of snapshots from an MD trajectory is shown in Fig. 4.
- Sample **frames** (snapshots) along the trajectory.
- Evaluate each frame using an electronic structure method (e.g., DFT, MP2, CCSD(T)) to obtain:

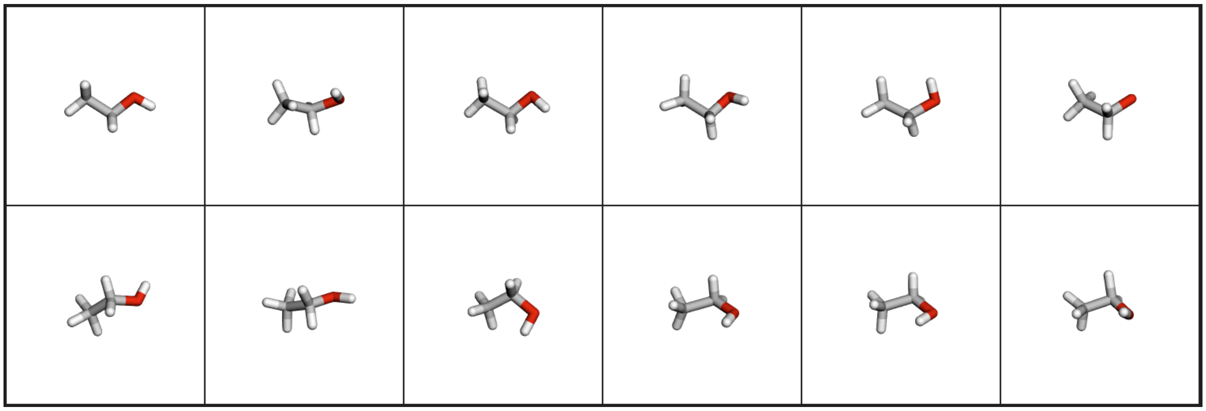

Figure 4: Molecular dynamics (MD) trajectory of ethanol, showing configurations sampled every 1000 steps. Data derived from the MD17 dataset.

1. total energy E_{tot}, and
2. atomic forces $\mathbf{F}_i = -\partial E_{\text{tot}} / \partial \mathbf{R}_i$.

Thus, the training dataset consists of $\{\mathbf{R}_i, E_{\text{tot}}, \mathbf{F}_i\}$ for each frame.

For small molecules, we can use precomputed datasets such as `MD17`, which contains ab-initio MD trajectories for several molecules. The dataset is available at `http://quantum-machine.org/gdml` and can be accessed via `PyTorch Geometric (PyG)`. PyG also provides many other molecular datasets: `https://pytorch-geometric.readthedocs.io/en/2.5.0/modules/datasets.html`.

Other datasets suitable for training machine learning potentials include:

- **ANI-1 dataset** [3], which contains millions of small organic molecules with DFT-calculated energies and forces.
- **QM9 dataset** [4], available via PyTorch Geometric (PyG), providing energies, geometries, and additional molecular properties for small organic molecules.
- **Open Catalyst Dataset (OC20/OC25)**, for catalysis applications, available at `https://github.com/Open-Catalyst-Project/Open-Catalyst-Dataset`, containing atomic configurations and energies for adsorbates on catalytic surfaces.

5 Lab

We will work on two labs: ACSF + FNN and SOAP + GPR:

Lab1: `https://colab.research.google.com/drive/1oKfvoXu6_gZyRk1-buEFd54T1vJOO_Gh?usp=sharing`

Lab2: https://colab.research.google.com/drive/161wJbOLj8dQIU93ISGdnRrQmkhRD_Cq_?usp=sharing

References

[1] Alea Miako Tokita and Jörg Behler. How to train a neural network potential. *The Journal of Chemical Physics*, 159(12), 2023.

[2] Albert P. Bartók, Risi Kondor, and Gábor Csányi. On representing chemical environments. *Phys. Rev. B*, 87:184115, May 2013.

[3] Justin S Smith, Olexandr Isayev, and Adrian E Roitberg. Ani-1, a data set of 20 million calculated off-equilibrium conformations for organic molecules. *Scientific data*, 4(1):170193, 2017.

[4] Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole Von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data*, 1(1):1–7, 2014.