nanoHUB Tutorial 06/16/2021

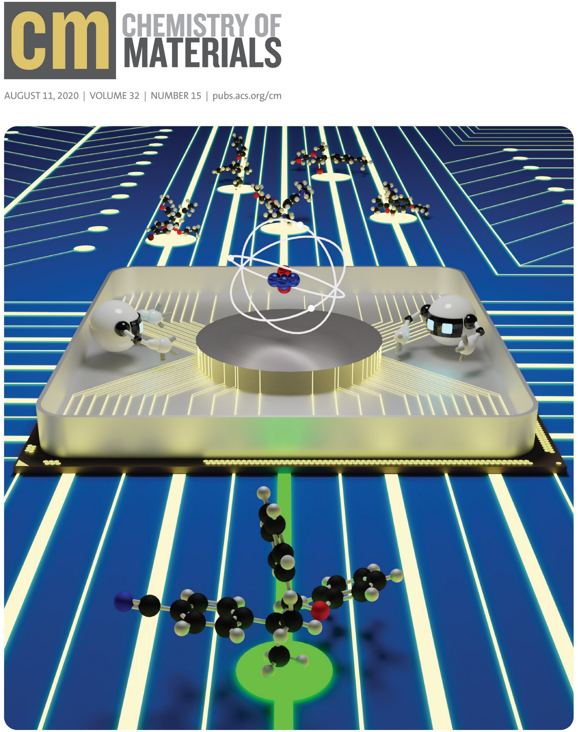
Argonne
NATIONAL LABORATORY

Active Learning via Bayesian Optimization for Discovery of Energy Storage Materials

Hieu A. Doan

Garvit Agarwal

Molecular Materials Group
Materials Science Division



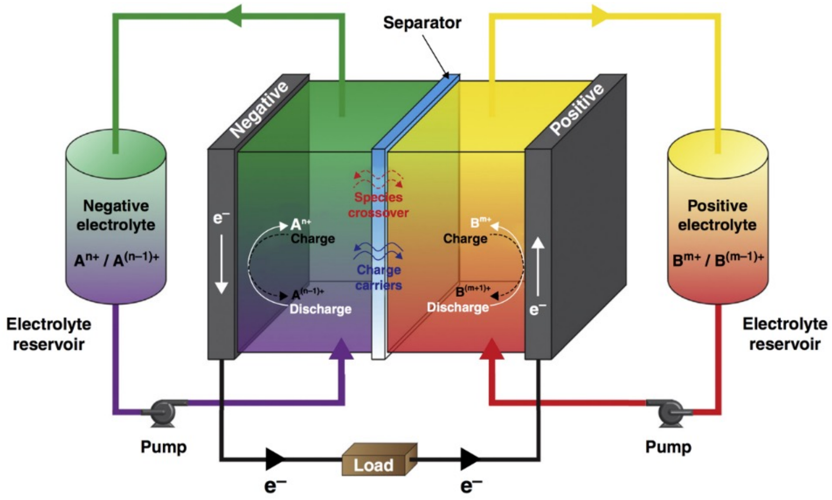
cm CHEMISTRY OF MATERIALS

AUGUST 11, 2020 | VOLUME 32 | NUMBER 15 | pubs.acs.org/cm

Doan, Agarwal, Qian, Counihan, Rodríguez-López, Moore, & Assary. (2020). https://doi.org/10.1021/acs.chemmater.0c00768

U.S. DEPARTMENT OF ENERGY
Argonne National Laboratory is a
U.S. Department of Energy laboratory
managed by UChicago Argonne, LLC.

Redox Flow Battery (RFB) as a Stationary Energy Storage System

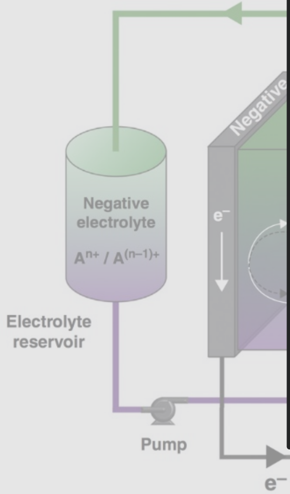


Kowalski, Su, Milshtein, Brushett (2016) *Current Opinion in Chemical Engineering*

Advantages:

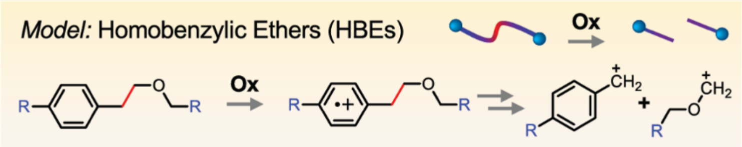
- ❑ Separation of energy and power capacity
- ❑ Integration of intermittent renewable energy sources into the electrical grid
- ❑ Non-aqueous RFBs potentially yield high energy density and lower cost

Redox Flow Battery (RFB) as a Stationary Energy Storage System



Kowalski, Su, Milshtein, Brushett (2016) *Current Opinion in Chemical Engineering*

Model: Homobenzylic Ethers (HBEs)



Desirable HBEs' properties:

- Suitable redox potential windows
- High solubility
- Ease of synthesis
- Electrochemical reversibility

s:

tion of energy and
capacity

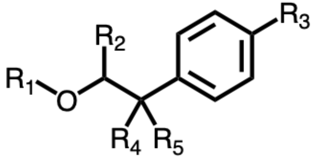
ion of intermittent
ble energy sources into
ctrical grid

ueous RFBs potentially
gh energy density and
ost

The Challenges

1. Large space of molecular candidates

HBE scaffold



∼

SMILES (Simplified Molecular-Input Line Entry System)

[**R3**]C1=CC=C(C([**R4**])([**R5**])C([**R2**])-[O][**R1**])C=C1

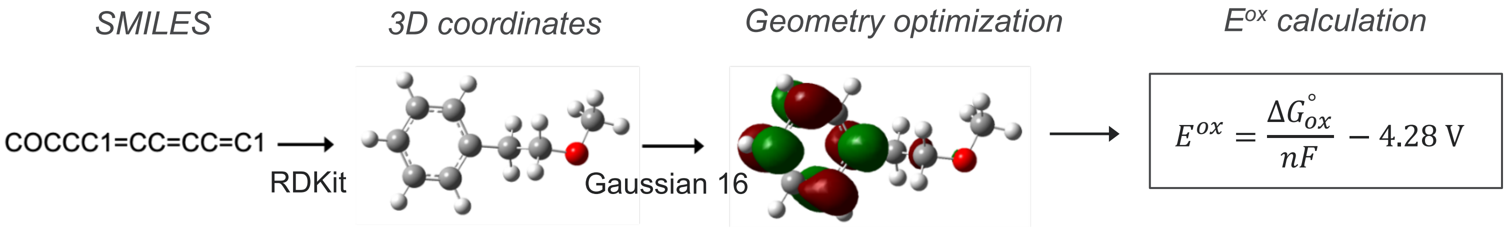
R$_1$ = -Me, -Et, -Pr, -Ph, -CN, -Eth, -COMe, -C(Me)Me, -CCOMe
R$_2$ = -N(Me)$_3^+$, -COMe, -Et, -OCMe, N(Me)$_2$, -NO$_2$, -C(=O), -Pr, -C(Me)Me, -CCOMe, -C(Me)OMe
R$_3$ = -N(Me)$_3^+$, -Me, -COMe, -Br, -C(=O), -OEth, -Pr, -C(Me)Me, -C(COMe), -C(Me)(OMe)
R$_4$ - R$_5$ = -N(Me)$_3^+$, -OMe, -COMe, -Br, -N(Me)Me, -Et, -OEt, -NO$_2$, -C(=O), -Pr, -C(Me)Me, -C(COMe), -C(Me)(OMe)

> 10^5 molecules

2. Expensive/time-consuming synthesis and characterization

The Opportunities

1. Density Functional Theory (DFT) calculations starting from SMILES



$$\boxed{E^{ox} = \frac{\Delta G^{\circ}_{ox}}{nF} - 4.28\ \mathrm{V}}$$

2. Train machine learning (ML) models using DFT-computed E^{ox}

Need ML models to not only make accurate predictions but also guide the selection of training data →

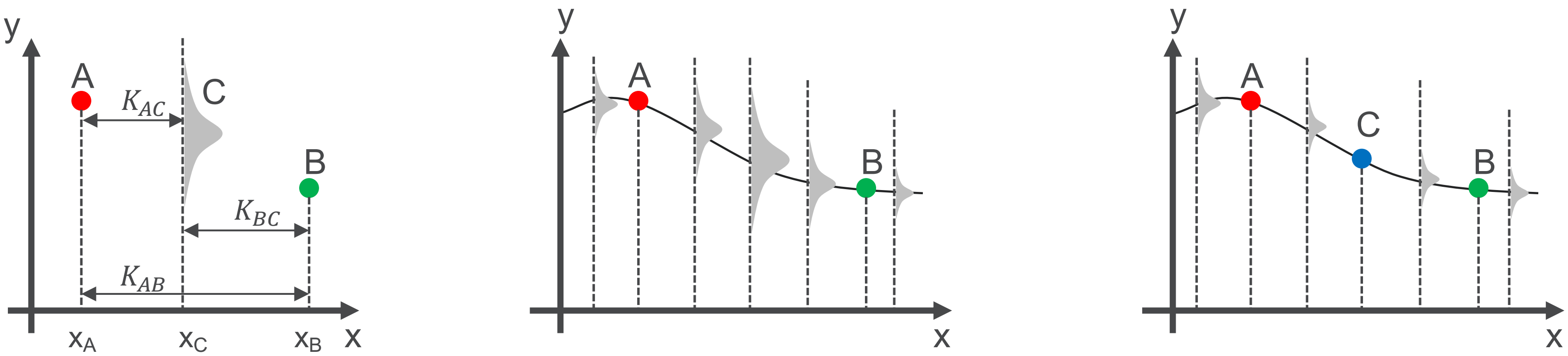
Active learning/ Bayesian optimization

(Surrogate model + Acquisition function)

Gaussian Process Regression (GPR) as a surrogate model

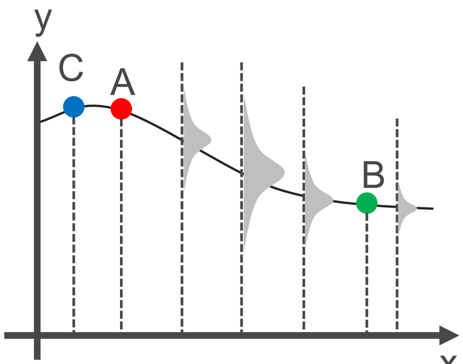
In a nutshell: Predict properties/outputs based on feature/input differences (distances)

Covariance is calculated as a function of (feature) distances, e.g. $\boxed{\mathrm{K}(x_1, x_2) = exp\left[-\frac{1}{2}\frac{(x_1-x_2)^2}{l^2}\right]}$

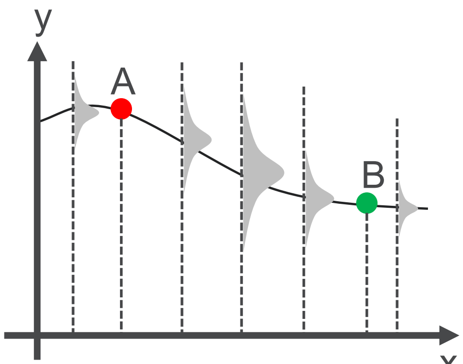


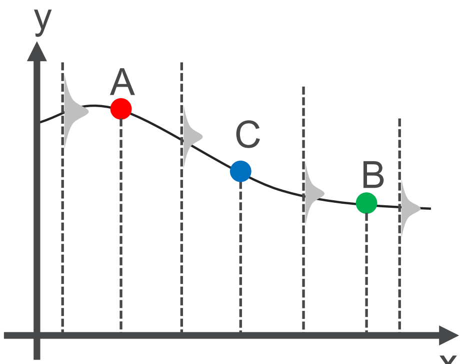
Use of GPR in Active learning/Bayesian optimization

GPR-predicted μ(x) and σ(x) enables **Active Learning/Bayesian optimization**



"Exploitation": follow $\mu(x)_{max}$

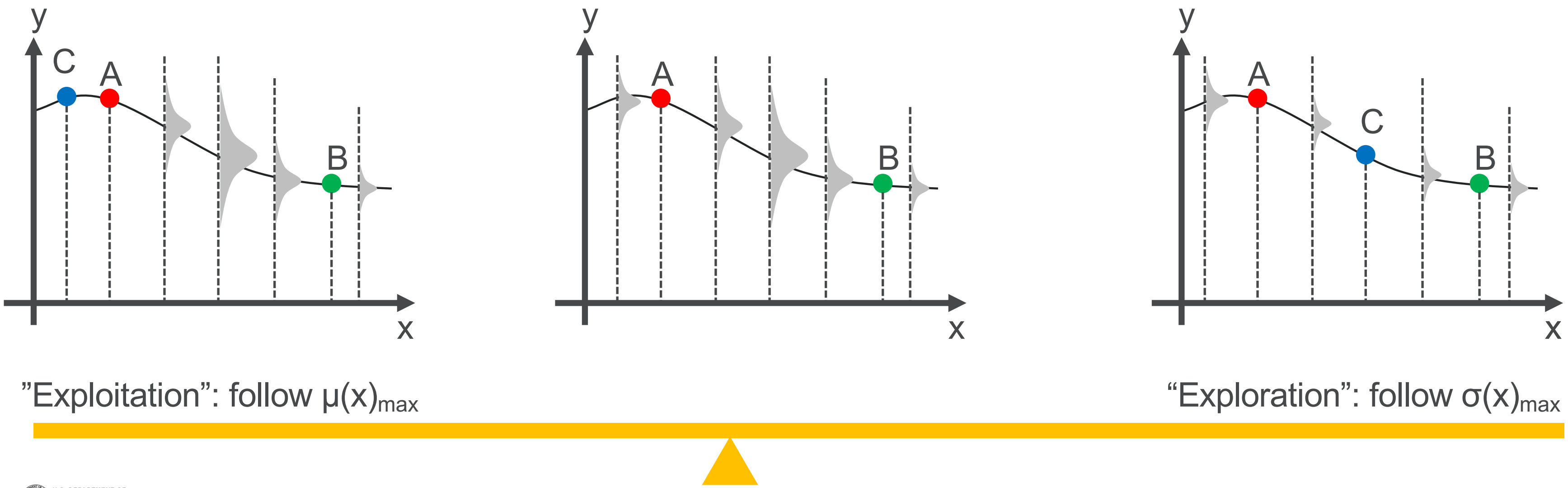




"Exploration": follow $\sigma(x)_{max}$

Acquisition function

Acquisition function formulates an optimal strategy toward an objective by guiding the next evaluation



”Exploitation”: follow $\mu(x)_{max}$

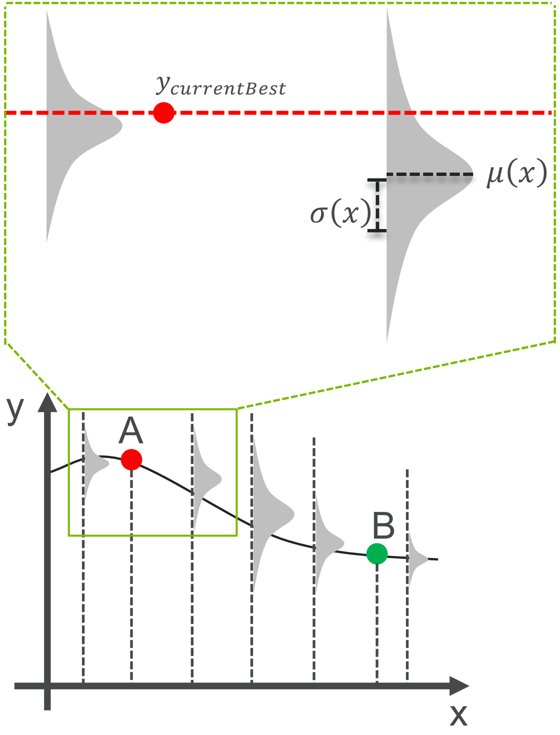
“Exploration”: follow $\sigma(x)_{max}$

Upper Confidence Bound (UCB)

Upper Confidence Bound (UCB)

$$UCB(x) = \mu(x) + \xi * \sigma(x)$$

$$x_{next} = argmax(UCB(x))$$



$\mu(x)$: Mean
$\sigma(x)$: Standard deviation

Probability of improvement (PI) and Expected improvement (EI)

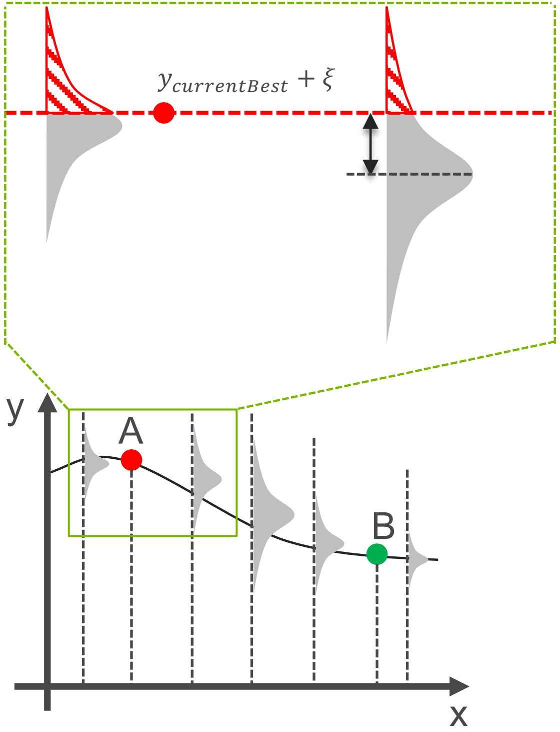
Probability of Improvement (PI)

$$PI(x) = P\left(y'_x \geq y_{currentBest} + \xi\right)$$

$$= \Phi(Z)$$

$$\mathrm{Z} = \frac{\mu(x) - y_{currentBest} - \xi}{\sigma(x)}$$

$$x_{next} = argmax(PI(x))$$



Expected Improvement(EI)

$$EI(x) =$$
$$\begin{cases} (\mu(x) - y_{currentBest} - \xi)\Phi(Z) + \sigma(x)\phi(x), & \sigma(x) > 0 \\ 0, & \sigma(x) = 0 \end{cases}$$

$$x_{next} = argmax(EI(x))$$

$\mu(x)$: Mean

$\sigma(x)$: Standard deviation

Φ: Cummulative Distribution Function

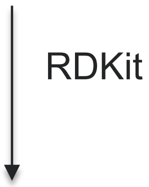
ϕ: Probability Density Function

Problem definition for Bayesian optimization

Input

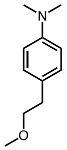
Molecule candidates
(SMILES library)

“SMILES.csv”



Feature vectors

E.g.



MW	# of C	# of aromatic rings	…
179	**11**	**1**	**…**

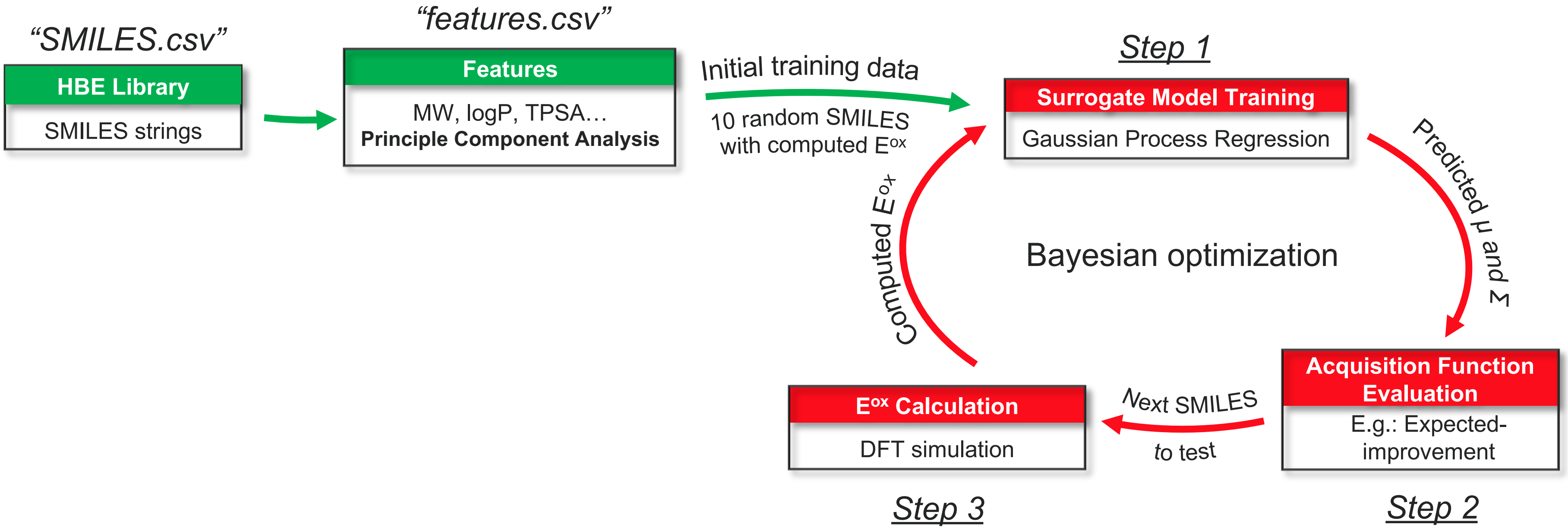
“features.csv” 125-component vectors

Output/Objective

Molecules w/ desired oxidation potential

Bayesian optimization scheme

Objective: Given 1000 HBE molecules, find one with maximized E^{ox} within N evaluations



"SMILES.csv"

HBE Library

SMILES strings

"features.csv"

Features

MW, logP, TPSA…

Principle Component Analysis

Initial training data

10 random SMILES with computed E^{ox}

Step 1

Surrogate Model Training

Gaussian Process Regression

Predicted μ and Σ

Bayesian optimization

Step 2

Acquisition Function Evaluation

E.g.: Expected-improvement

Next SMILES to test

Step 3

E^{ox} Calculation

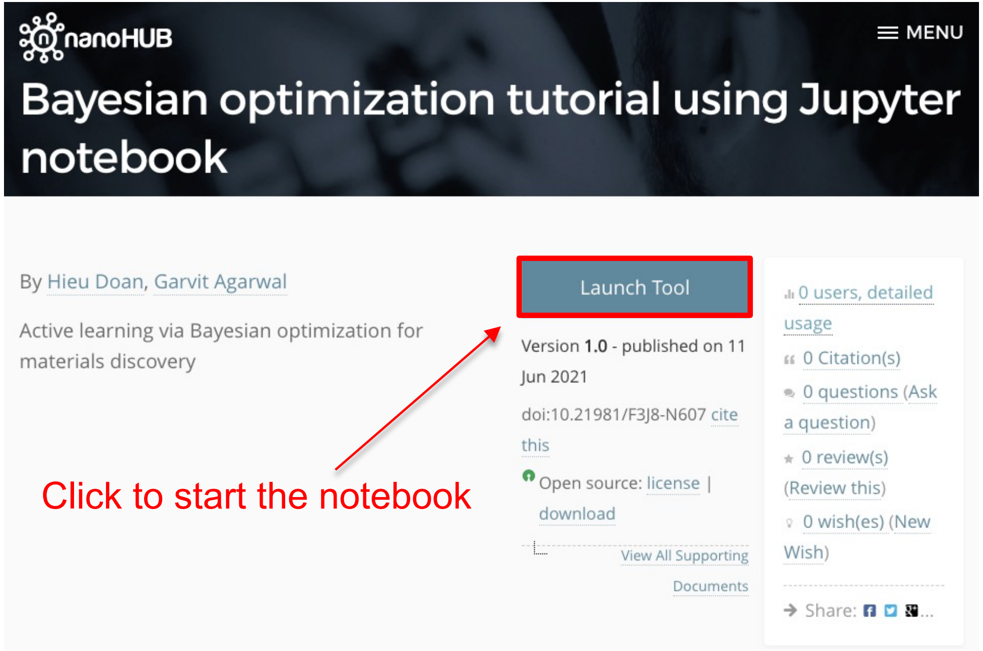
DFT simulation

Computed E^{ox}

Code adapted from https://github.com/rajak7/Bayesian_Optimization_Material_design

Launch the nanoHUB tool

From your browser go to this link: https://nanohub.org/resources/bayesopt



nanoHUB

≡ MENU

Bayesian optimization tutorial using Jupyter notebook

By Hieu Doan, Garvit Agarwal

Active learning via Bayesian optimization for materials discovery

Version 1.0 - published on 11 Jun 2021

doi:10.21981/F3J8-N607 cite this

Open source: license | download

View All Supporting Documents

0 users, detailed usage

0 Citation(s)

0 questions (Ask a question)

0 review(s) (Review this)

0 wish(es) (New Wish)

Share:

Click to start the notebook