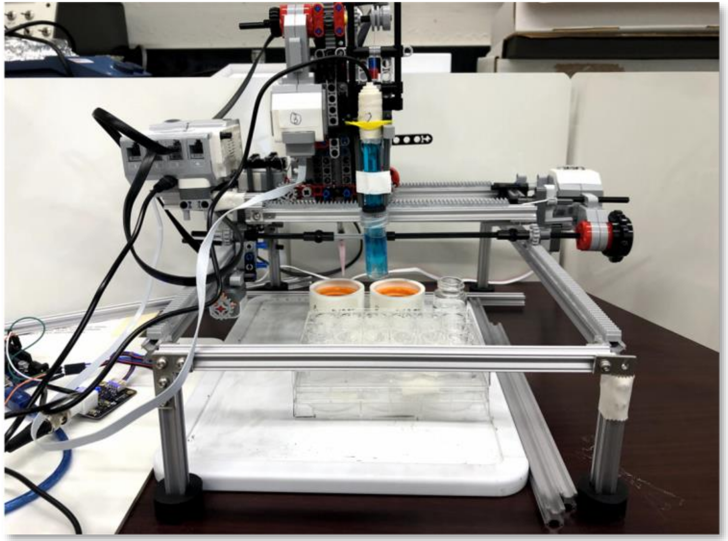
A Low-Cost Education Platform for Teaching Autonomous Physical Science



Logan Saar

*Senior - Materials Science and Engineering, UMD (**lsaar@umd.edu**)*

Ichiro Takeuchi, Gilad Kusne, Austin McDannald

University of Maryland & NIST

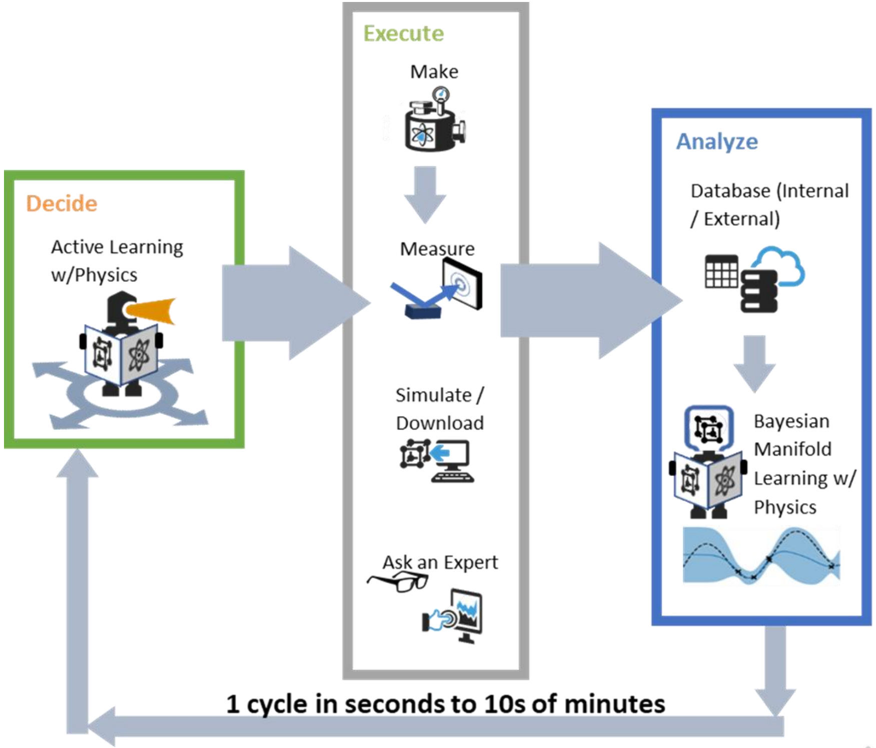
CAMEO: Closed-Loop Autonomous Materials Exploration and Optimization

Discovered: New best-in-class phase change memory material

ScientificAI: built in phase map and XRD physics

10x acceleration over off-the-shelf methods

Run at: SLAC

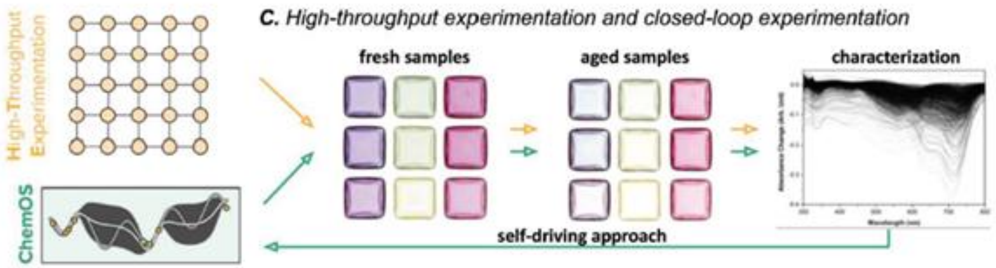


Kusne, et al. Nature Communications 11.1 (2020)

A mobile robotic chemist

Burger et al., Nature **583**, 237 (2020)

Beyond Ternary OPV: High-Throughput Experimentation and Self-Driving Laboratories Optimize Multicomponent Systems



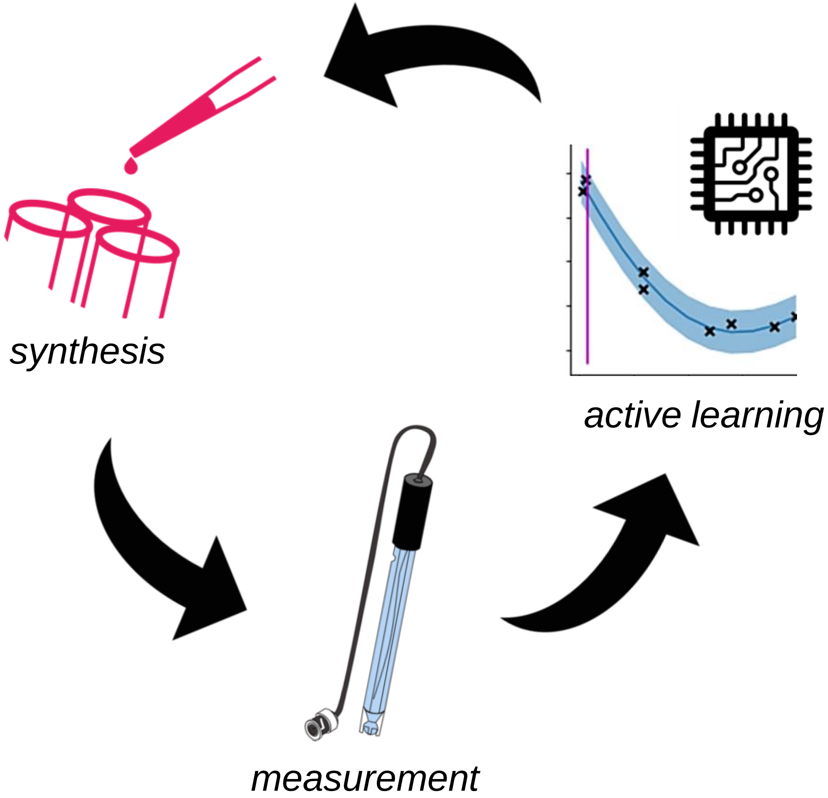
- Blending/mixing or polymers/organic molecules
- Number of experiments can be significantly reduced

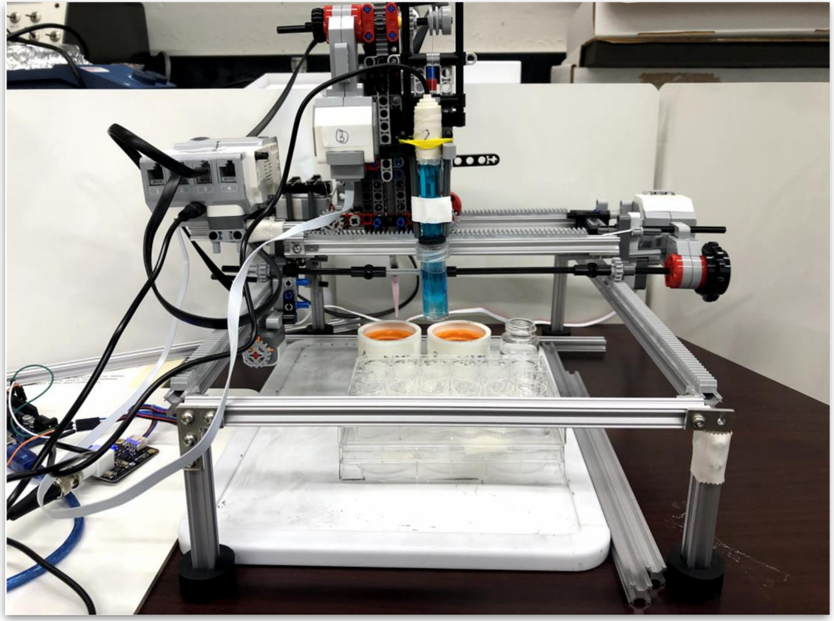
Burger, B., Maffettone, P.M., Gusev, V.V. et al. A mobile robotic chemist. Nature 583, 237–241 (2020)

Other Works

Stach, E., DeCost, B., Kusne, A. G., Hattrick-Simpers, J., Brown, K. A., Reyes, K. G., Schrier, J., Billinge, S., Buonassisi, T., Foster, I., Gomes, C. P., Gregoire, J. M., Mehta, A., Montoya, J., Olivetti, E., Park, C., Rotenberg, E., Saikin, S. K., Smullin, S., … Maruyama, B. (2021). Autonomous experimentation systems for materials development: A community perspective. Matter, 4(9), 2702-2726. https://doi.org/10.1016/j.matt.2021.06.036

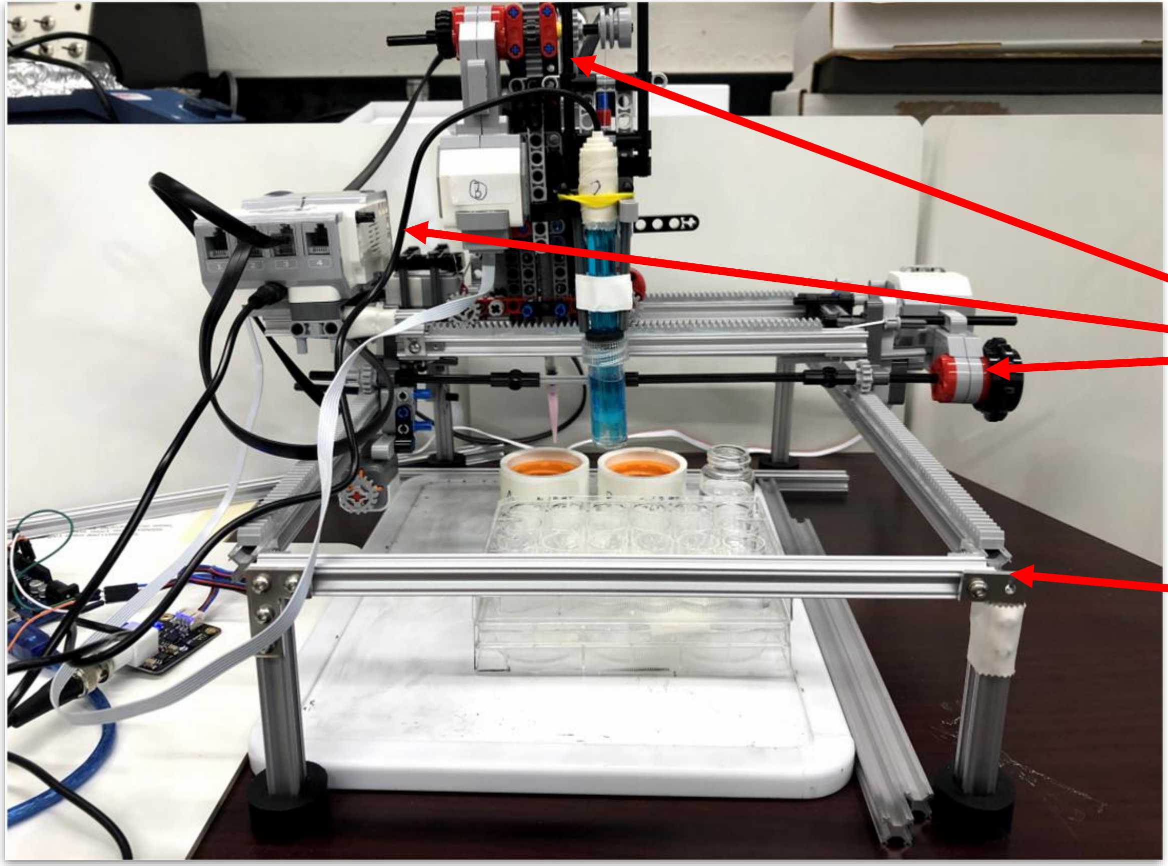
Low Cost Autonomous Physical Science System





< $400

Construction

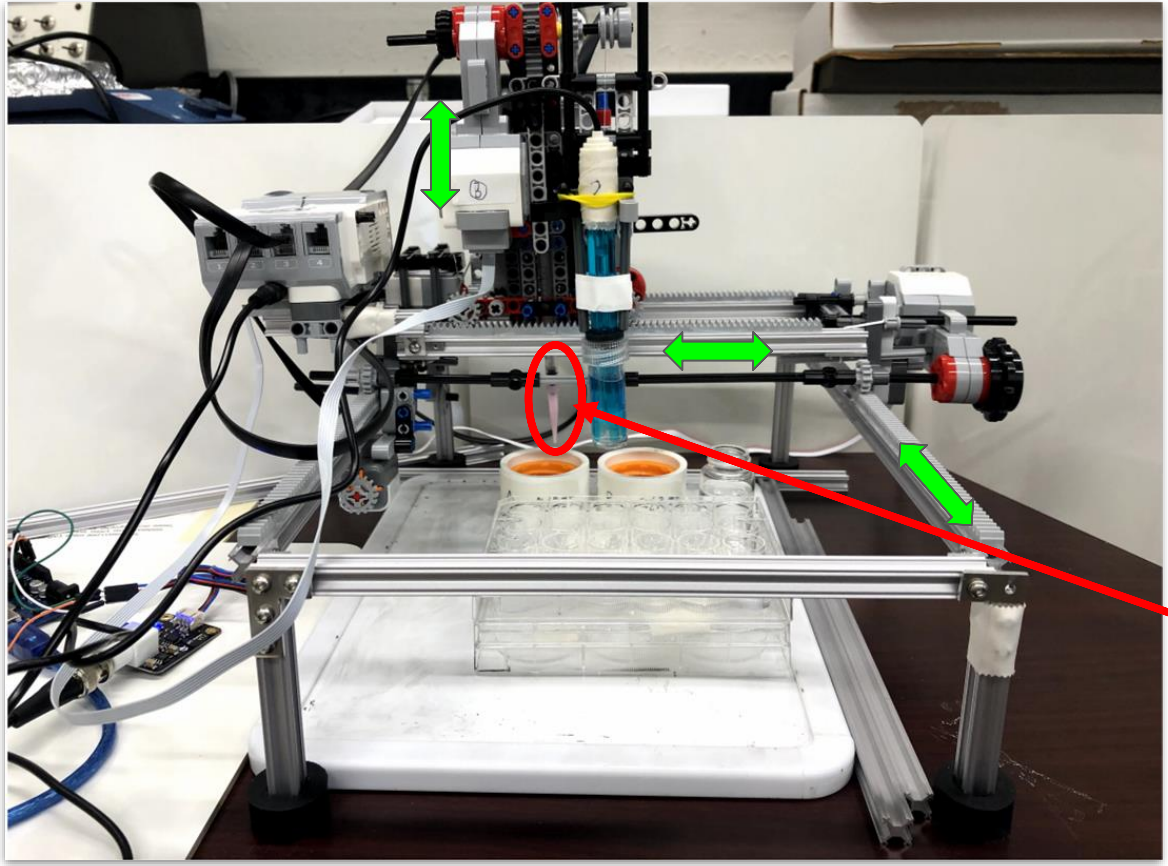


Lego Mindstorm Components

Aluminum Frame

Gerber, et al. (2017). Liquid-handling Lego robots and experiments for STEM education and research. PLOS Biology, 15(3)

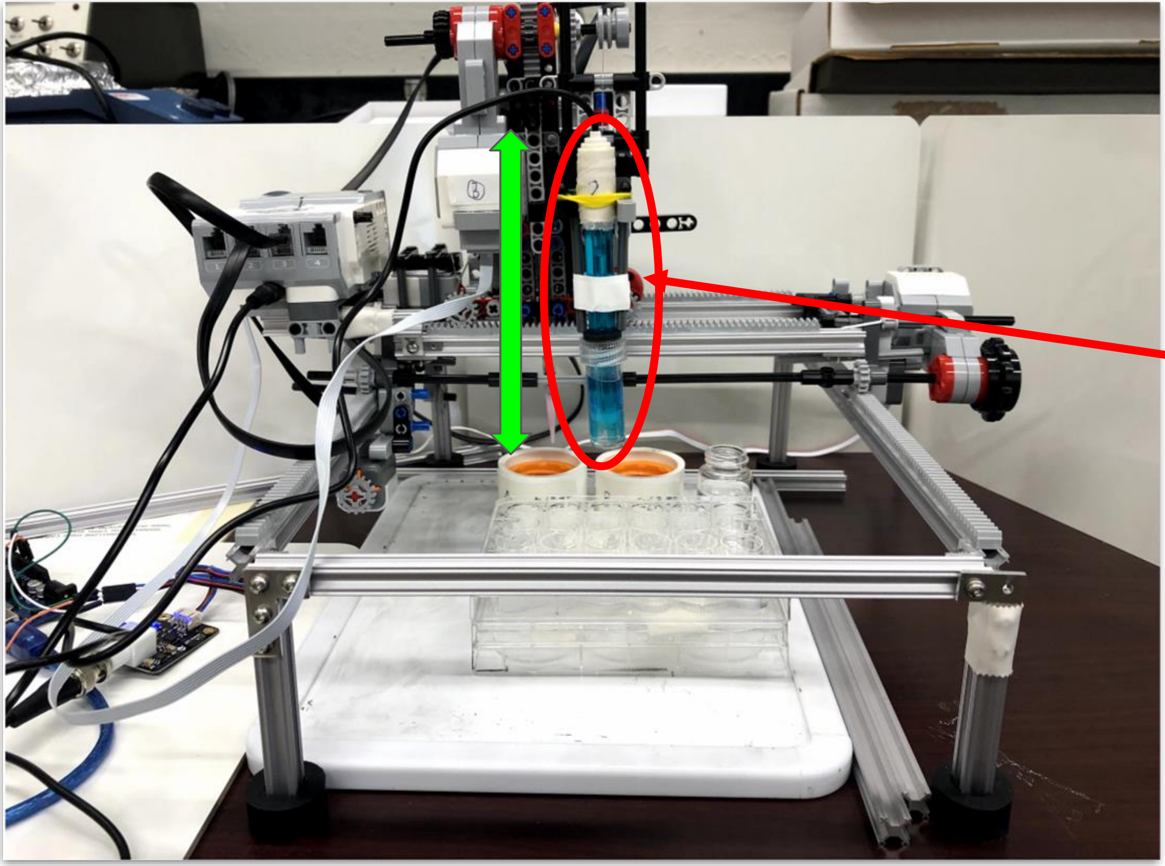
Synthesis Capabilities



X - Y - Z mobility

Syringe for sample collection / deposition

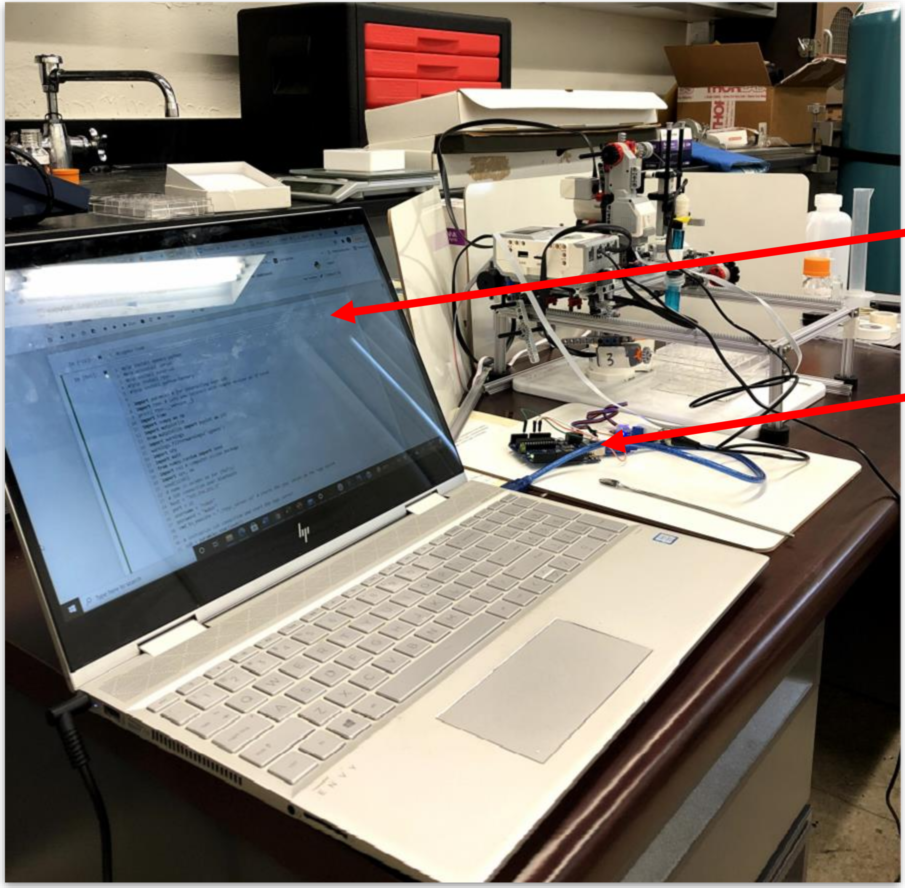
Measurement Capabilities



Arduino electrochemical
pH probe
(voltage readings)

$30

Setup and Control



Python Script

Arduino pH meter/USB

PC/Robot BT connection

Exploring pH of Buffer Solutions

Composition Space

Weak Acid - *Acetic Acid* - 1 M

Conjugate Base - *Sodium Acetate Solution* - 1 M

Goal

Recover Henderson-Hasselbalch Equation.

Henderson-Hasselbalch (HH) Equation:

$$\mathrm{pH} = \mathrm{p}K_{\mathrm{a}} + \log_{10}\left(\frac{[\mathrm{Base}]}{[\mathrm{Acid}]}\right)$$

Our known parameters (for synthesis)

Response Variable (measured)

Dissociation Constant (unknown to robot)

Active Learning Closed Loop System

https://www.youtube.com/watch?v=Ul9sx29vAXE&t=1s

Educational Application (Fall 2021 ENMA 437/637)

UMD Machine Learning for Materials Science Course

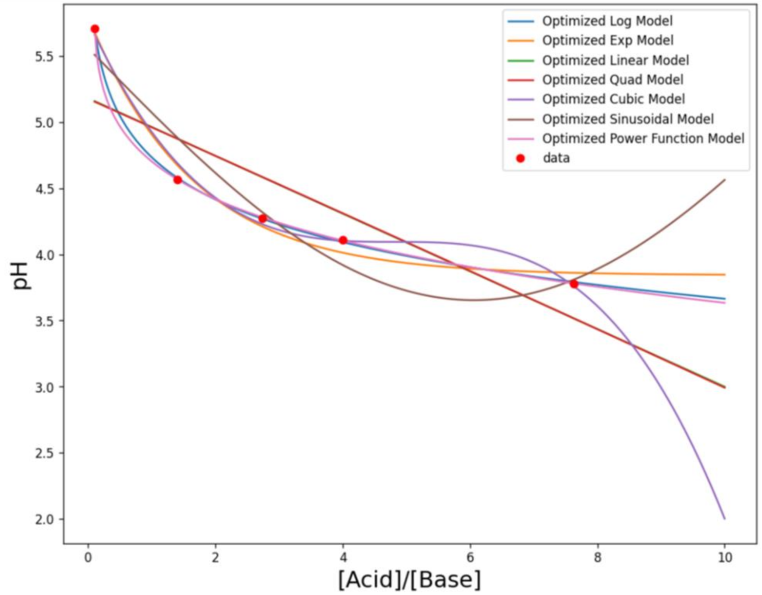
- 3 working systems
- Teams of 5 students
- Buffer Solution pH
  - Gaussian Process
  - Explore pH as f(composition)
- Iodine Clock

Model Determination

Can the Robot Determine the Physical Law by Itself?

Can the Robot Discover the Physical Law?

1. **Fit multiple functional forms to the data (“Candidates”)**
- (sinusoidal, power function, logarithmic, exponential, quadratic, etc.)
- Non-linear least squares regression



x = [Acid]/[Base]

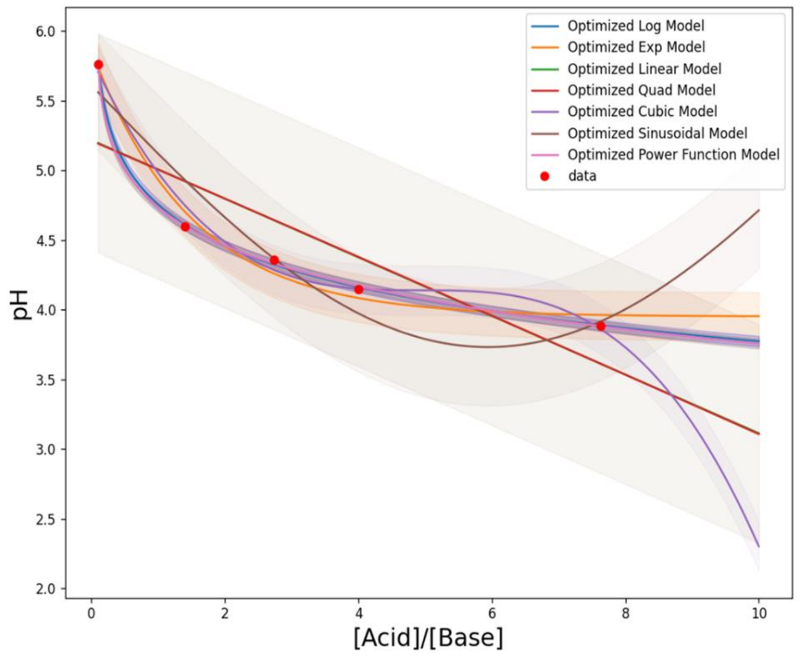
What is the correct form?

pH = A + B * **log** [C * (x-D)] ?
pH = A + B * **sin** [C * (x-D)] ?
pH = A + B * **exp**[C * (x-D)] ?
…

Alter parameters to get best fit

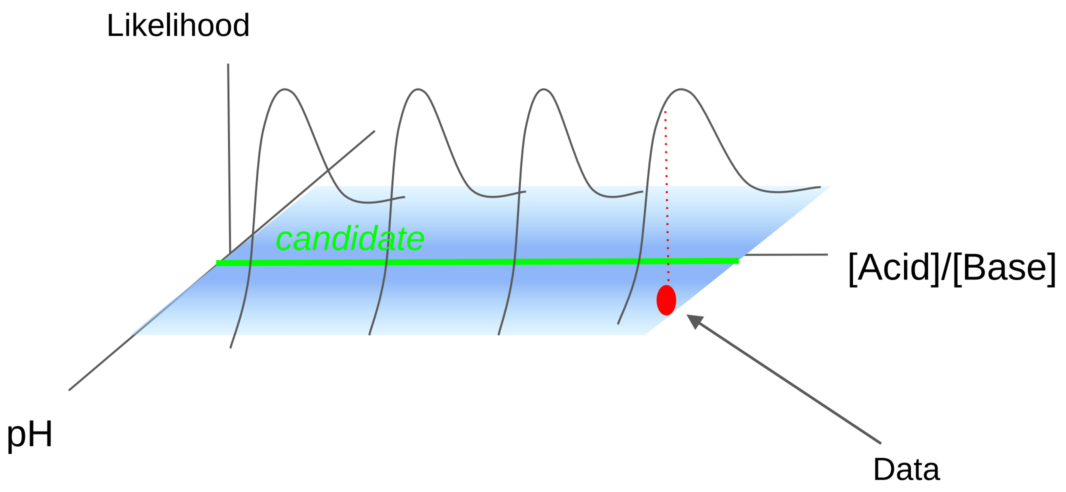
Can the Robot Discover the Physical Law?

2. Create PDF for each candidate at every composition
- (std. of PDF given by std. of residuals)
- Better models have narrow distributions, Worse are broad



Can the Robot Discover the Physical Law?

3. Rank the likelihood that each candidate model produced the data

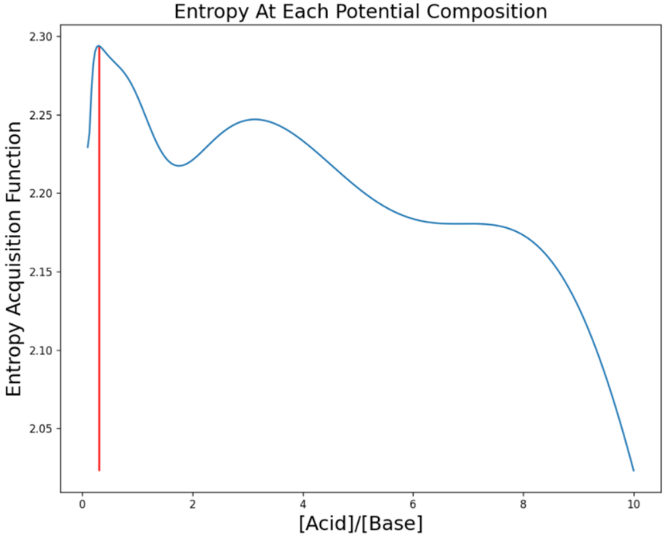


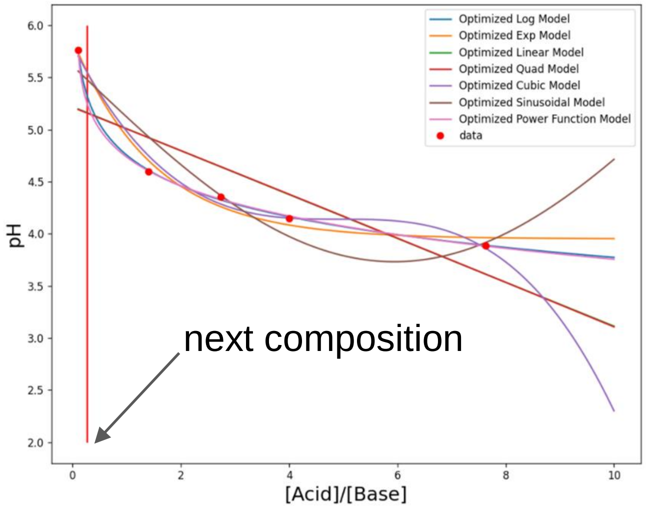
Performance Metric for each candidate is the sum of log(likelihood) along every collected data point

Yes, it did!

4. Determine which composition to measure next

- Look for composition where candidate predictions differ the most
- Weight “better” candidates more (Entropy of Cumulative Dist.)





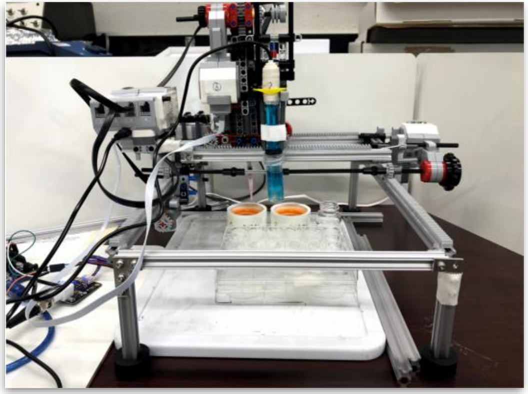
After 5 measurements:

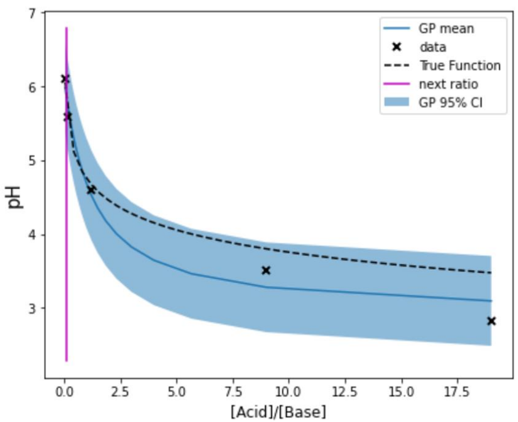
Top Ranked Model
pH = 4.753 + 1.02 * **log** [A/B]

Acetic Acid HH Equation:
pH = 4.756 + 1.00 * **log** [A/B]

Summary

Closed Loop Autonomous Science System
- ➔ Educational Tool
- ➔ Low-cost
- ➔ Modular
- ➔ Materials Exploration





Used Gaussian Processes to explore pH as function of composition
- ➔ Flexible model
- ➔ Explore other active learning methods ...

Acknowledgments

Dr. Ichiro Takeuchi, PhD - UMD

Dr. Gilad Kusne, PhD - NIST

Dr. Austin McDannald, PhD - NIST

Alex Wang - UMD

Haotong Liang – UMD

SURF Program – NIST

MRS

***Email** - lsaar@umd.edu*

References

Burger, B., Maffettone, P.M., Gusev, V.V. *et al.* A mobile robotic chemist. *Nature* 583, 237–241 (2020). https://doi.org/10.1038/s41586-020-2442-2

De Levie, R. (2003). The Henderson-hasselbalch equation: Its history and limitations. Journal of Chemical Education, 80(2), 146. https://doi.org/10.1021/ed080p146

Gerber, L. C., Calasanz-Kaiser, A., Hyman, L., Voitiuk, K., Patil, U., & Riedel-Kruse, I. H. (2017). Liquid-handling Lego robots and experiments for STEM education and research. PLOS Biology, 15(3), e2001413. https://doi.org/10.1371/journal.pbio.2001413

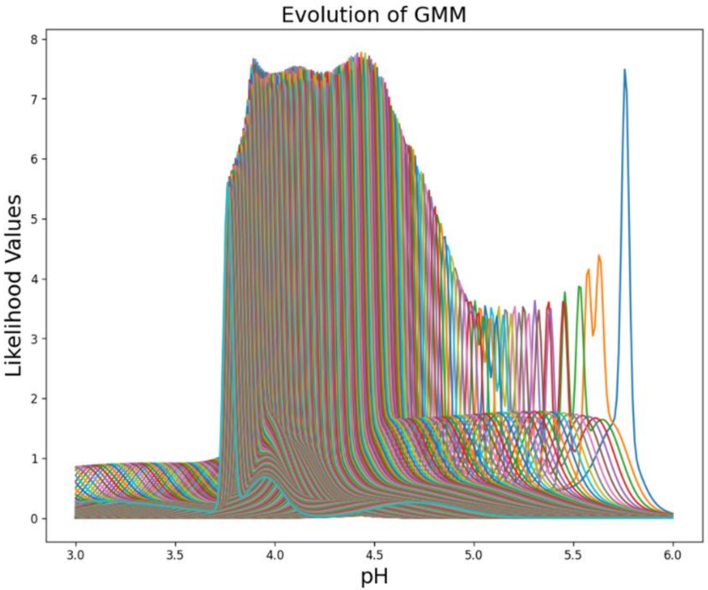
Questions

Appendix

Can the Robot Discover the Physical Law?

4. Calculate the Entropy of the GMM at every composition

- probability = (area under GMM likelihood curve)

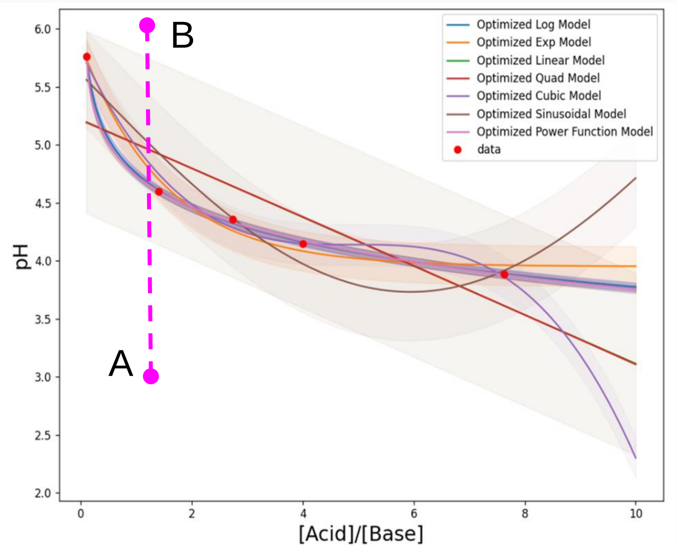


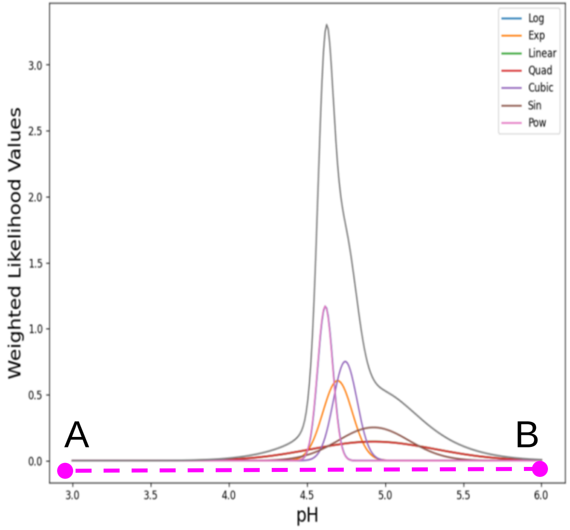
$$Entropy\ (S) \;=\; -\ \Sigma\,(p\ log\ p)$$

$$Probability\ (p) \;=\; \int_{pH_1}^{pH_2} (GMM)\,d(pH)$$

Can the Robot Discover the Physical Law?

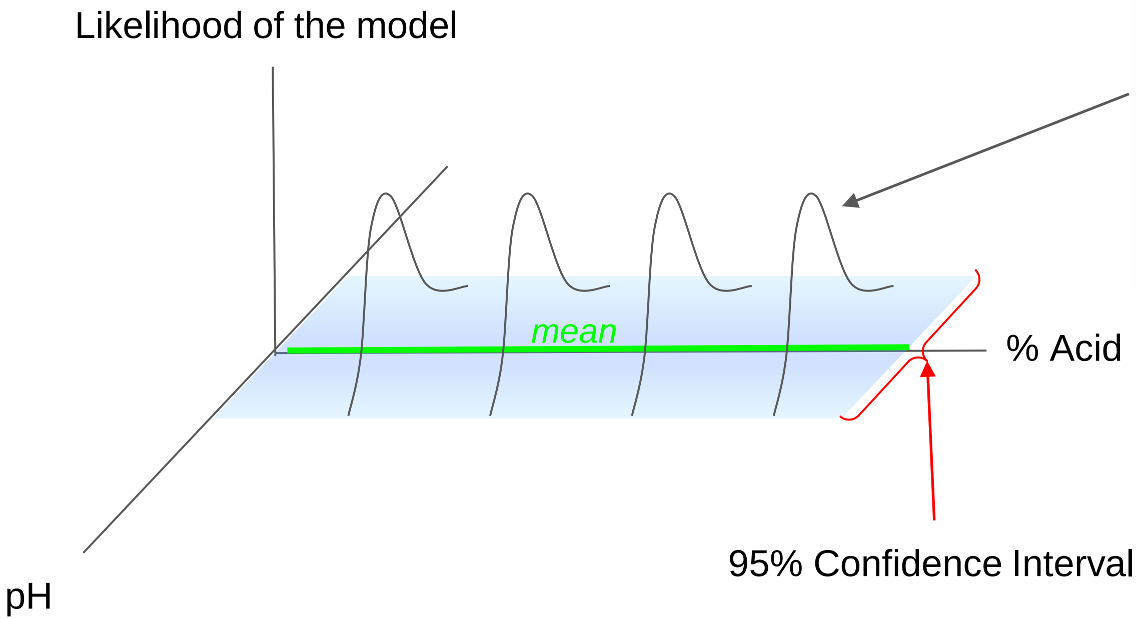
4. Create a cumulative distribution of all PDFs at each composition

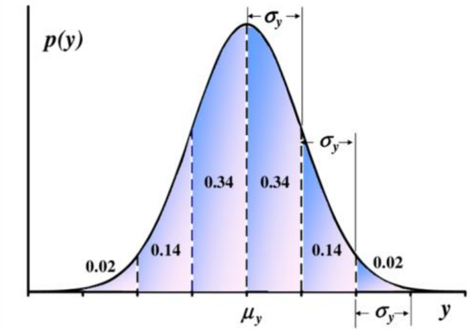




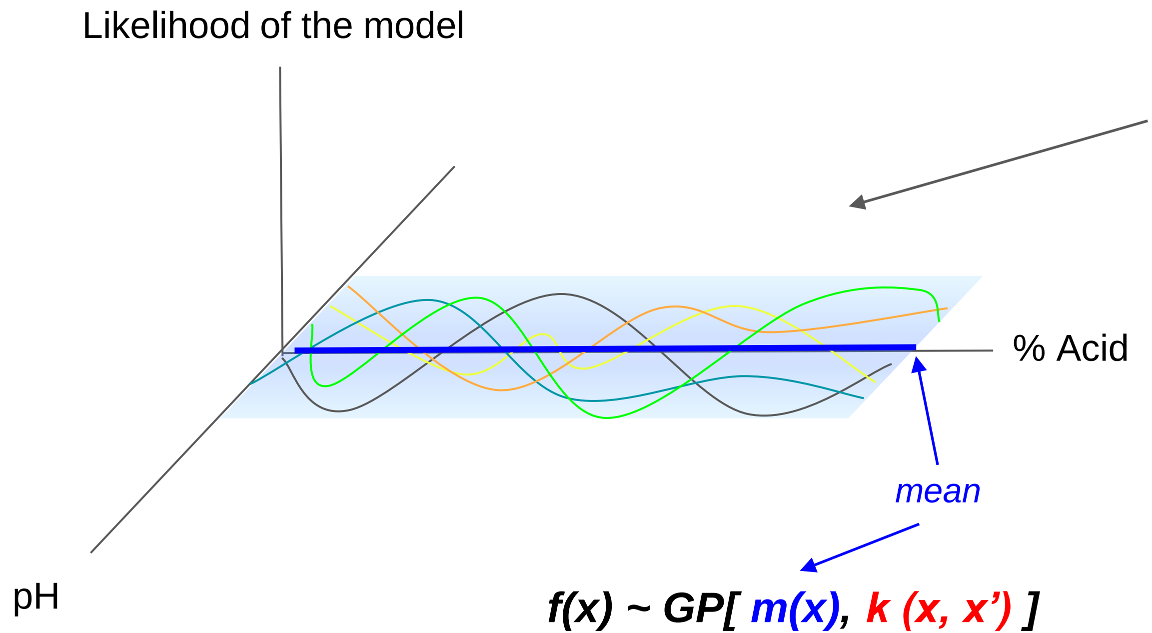
Exploration Initiative - **(Gaussian Process)**

Normal distribution of predicted pH for each potential % Acid composition





Exploration Initiative - (Gaussian Process)

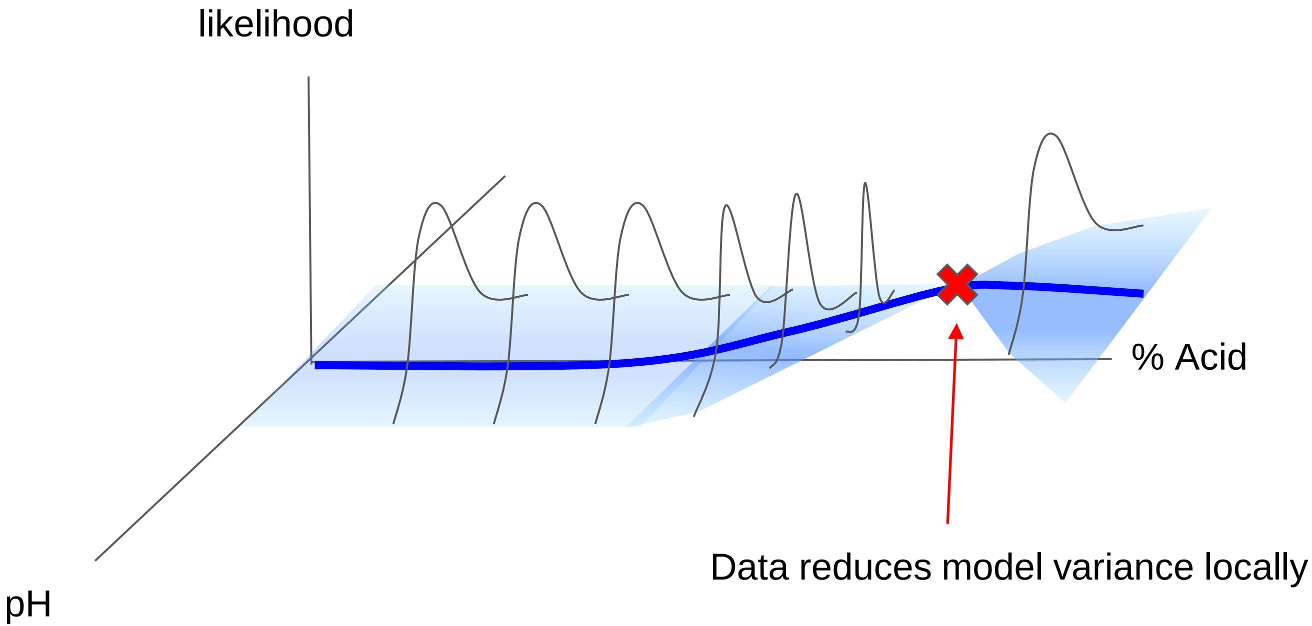


Many models generated:
- ➔ Non-parametric
- ➔ Flexible
- ➔ Averaged
- ➔ Kernel determines "smoothness"
- ➔ Noise tolerant

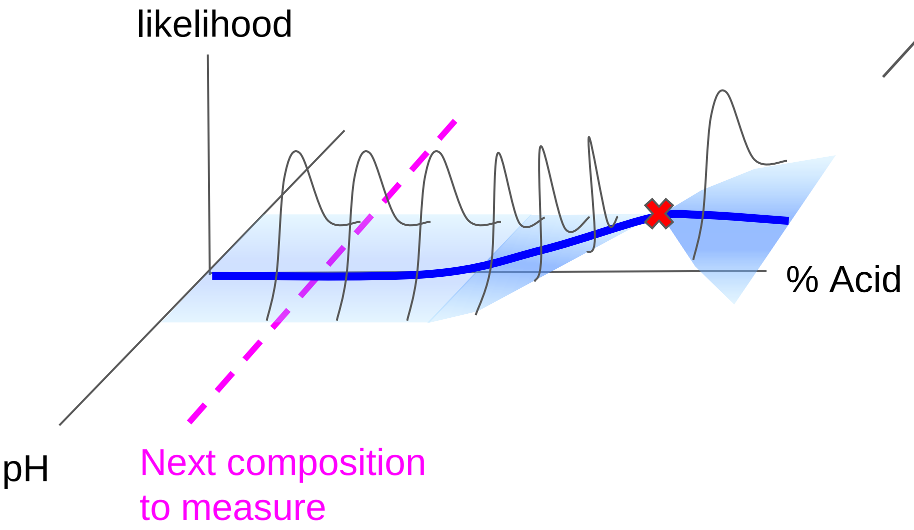
f(x) ~ GP[m(x), k (x, x')]

Kernel is a covariance function

Exploration Initiative - **(Gaussian Process)**

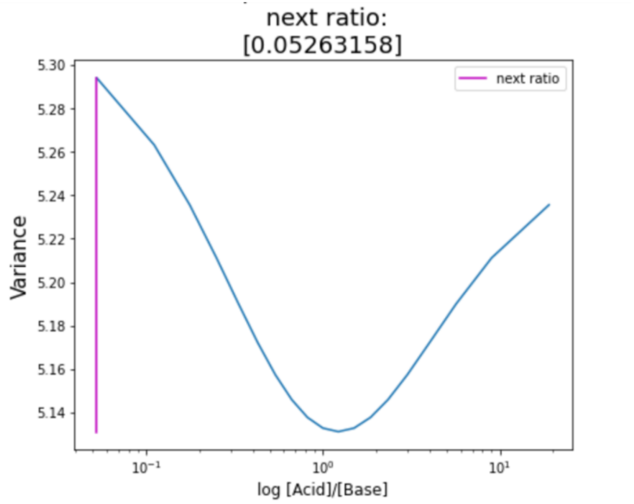


Exploration Initiative - **(Gaussian Process)**



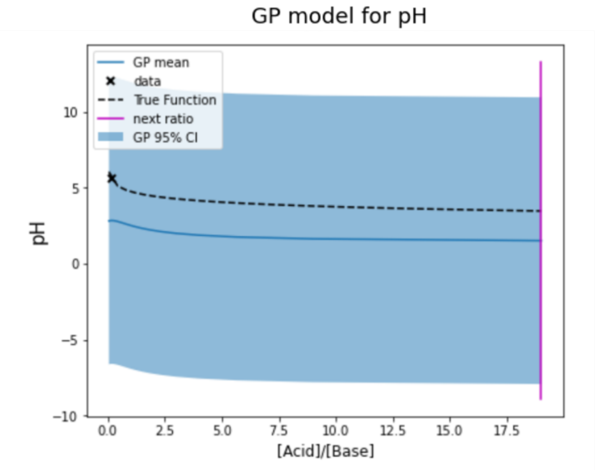
Active Learning:

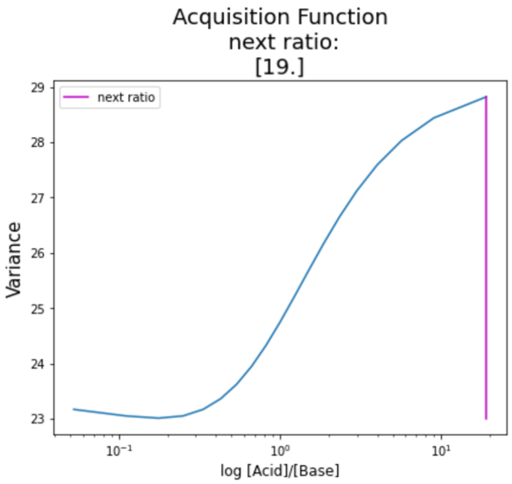
- ➔ Acquisition Function
- ➔ *argmax (variance)*
- ➔ *Optimization initiative*
- ➔ *Scheduling of initiatives*



Autonomous Results - <u>**(Gaussian Process)**</u>

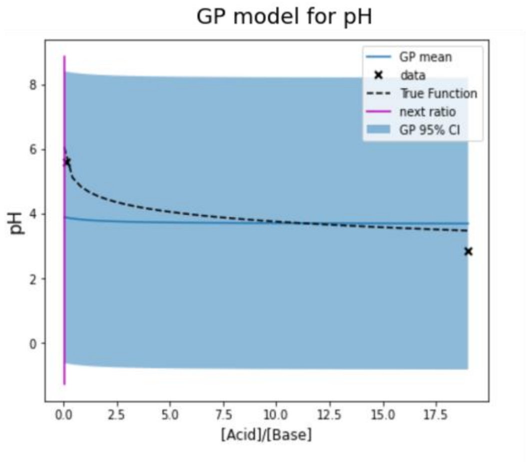
1 data point

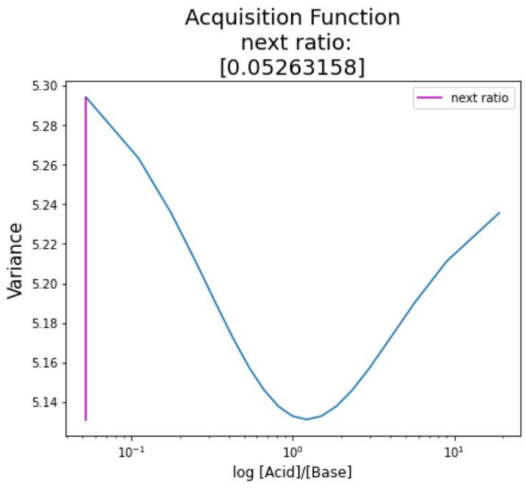




Autonomous Results - **(Gaussian Process)**

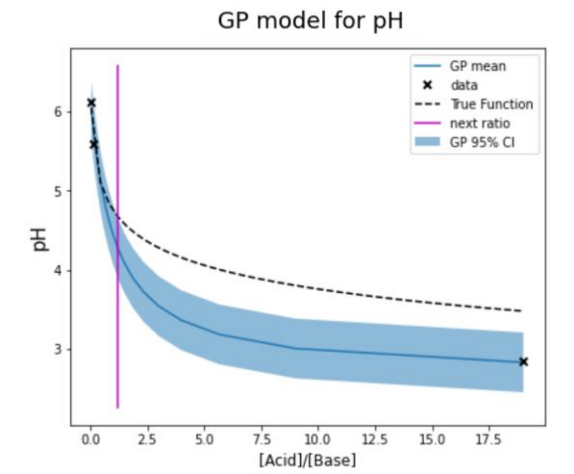
2 data points

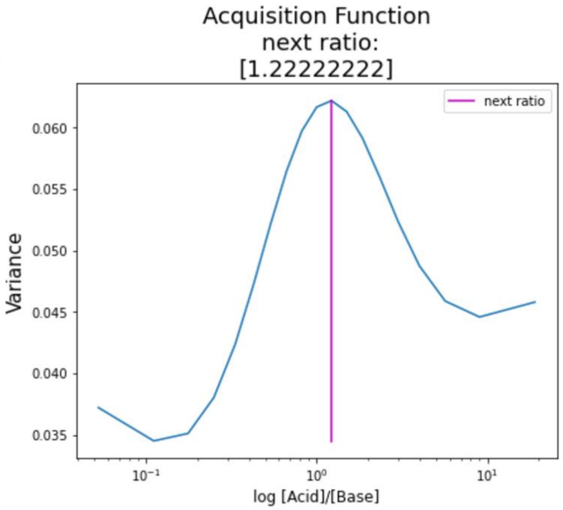




Autonomous Results - **(Gaussian Process)**

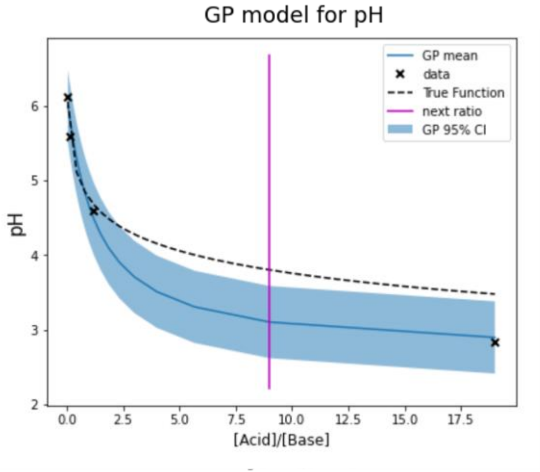
3 data points

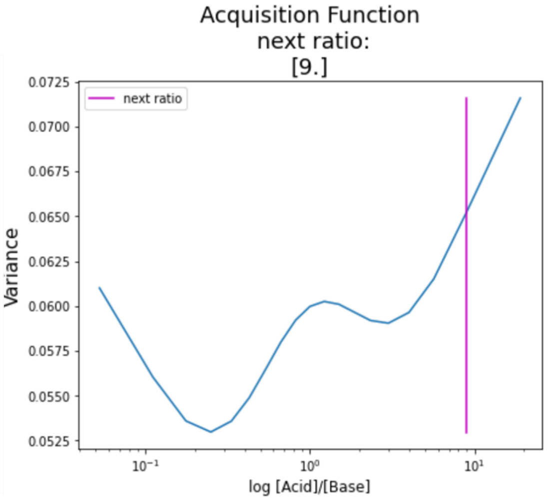




Autonomous Results - (Gaussian Process)

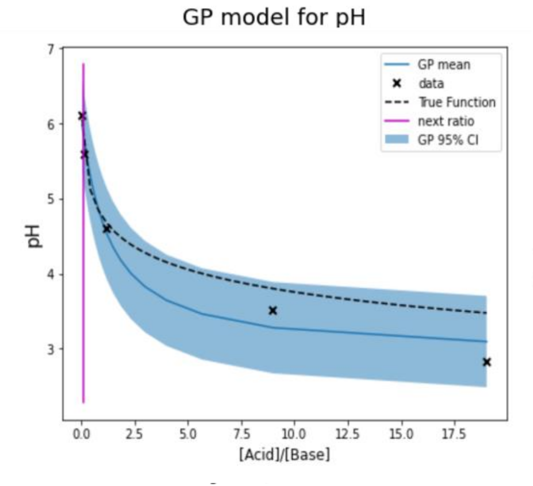
4 data points

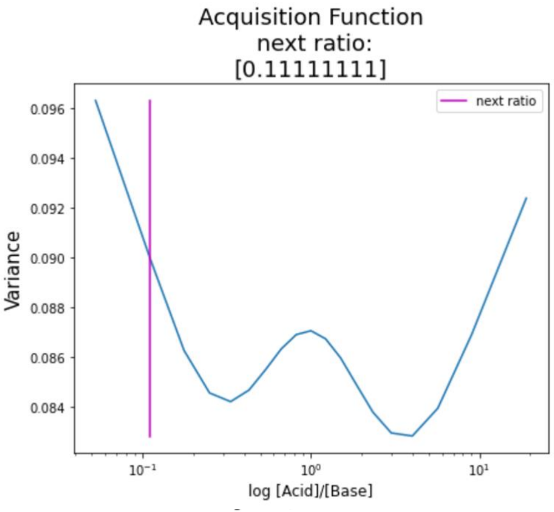




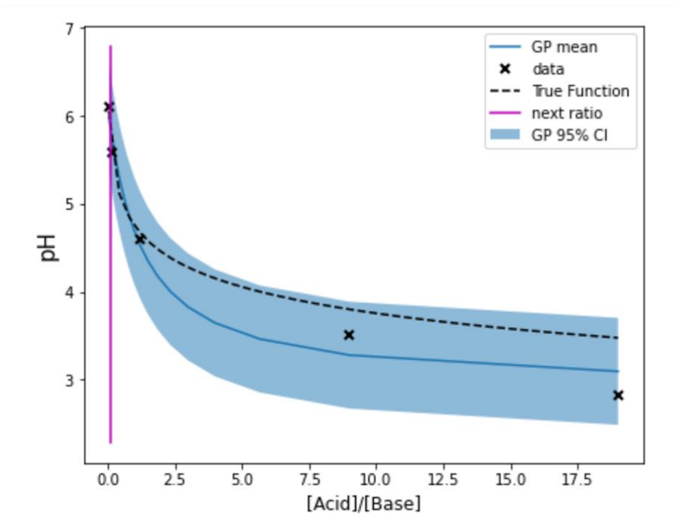
Autonomous Results - (Gaussian Process)

5 data points





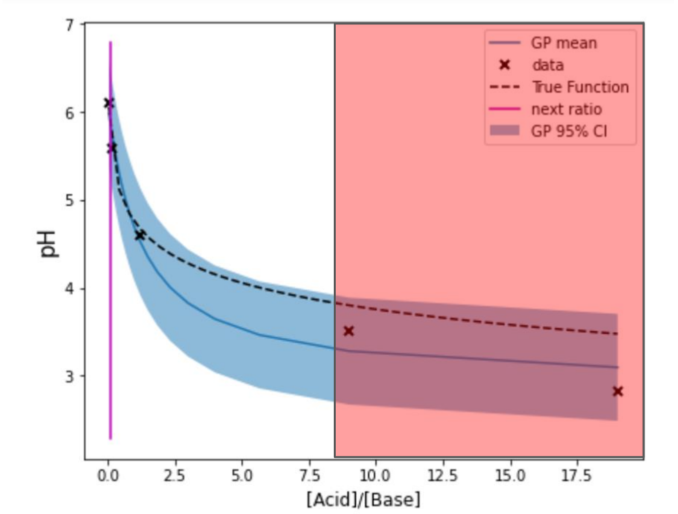
GP flexibility - **(Gaussian Process)**



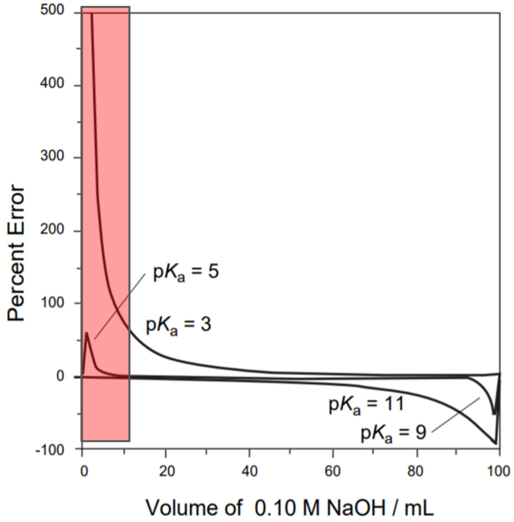
HH equation relies on assumptions

- ➔ No self-ionization of water
- ➔ Valid only in certain composition range
- ➔ Our pKa ~ 4.7

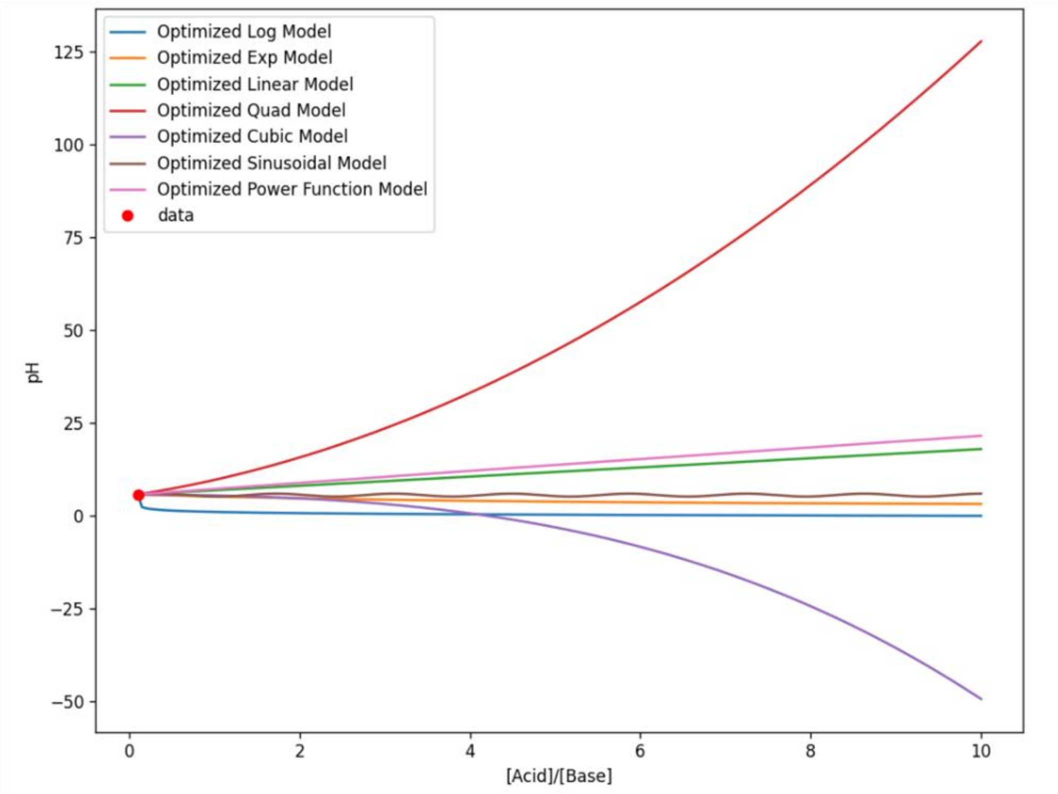
GP flexibility - **(Gaussian Process)**



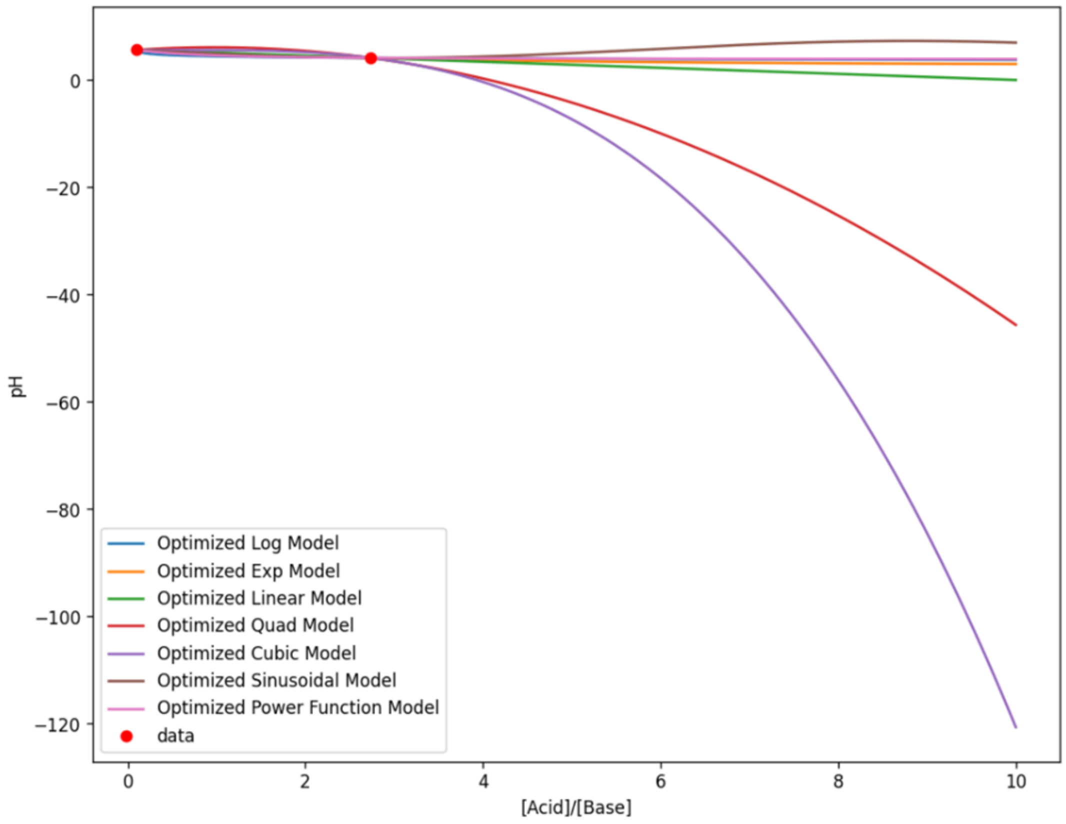
HH equation relies on assumptions
- ➔ No self-ionization of water
- ➔ Valid only in certain composition range
- ➔ pKa ~ 4.7

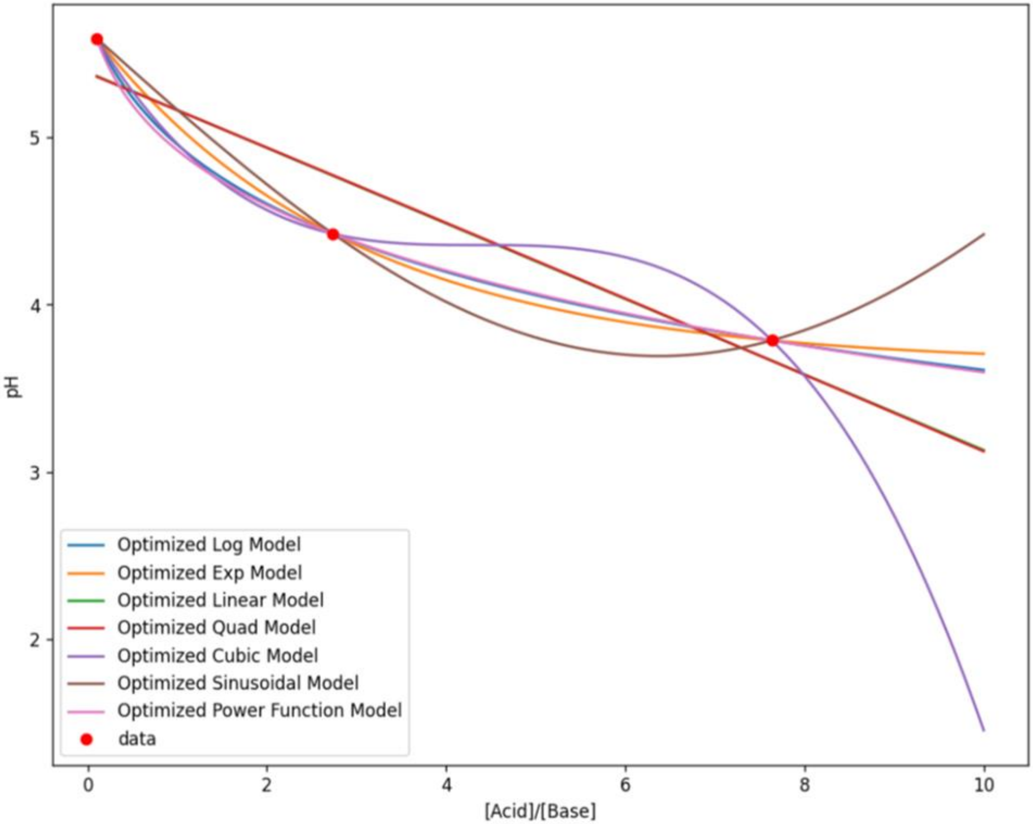


↑↑ % error in HH simplification ...

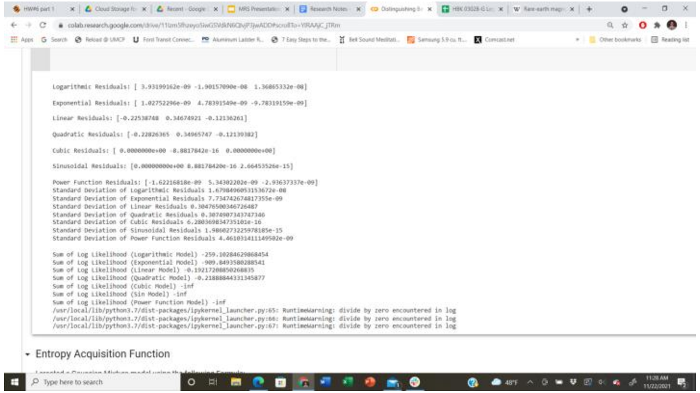


Optimized Log Model
Optimized Exp Model
Optimized Linear Model
Optimized Quad Model
Optimized Cubic Model
Optimized Sinusoidal Model
Optimized Power Function Model
data
pH
[Acid]/[Base]
125
100
75
50
25
0
−25
−50
0
2
4
6
8
10

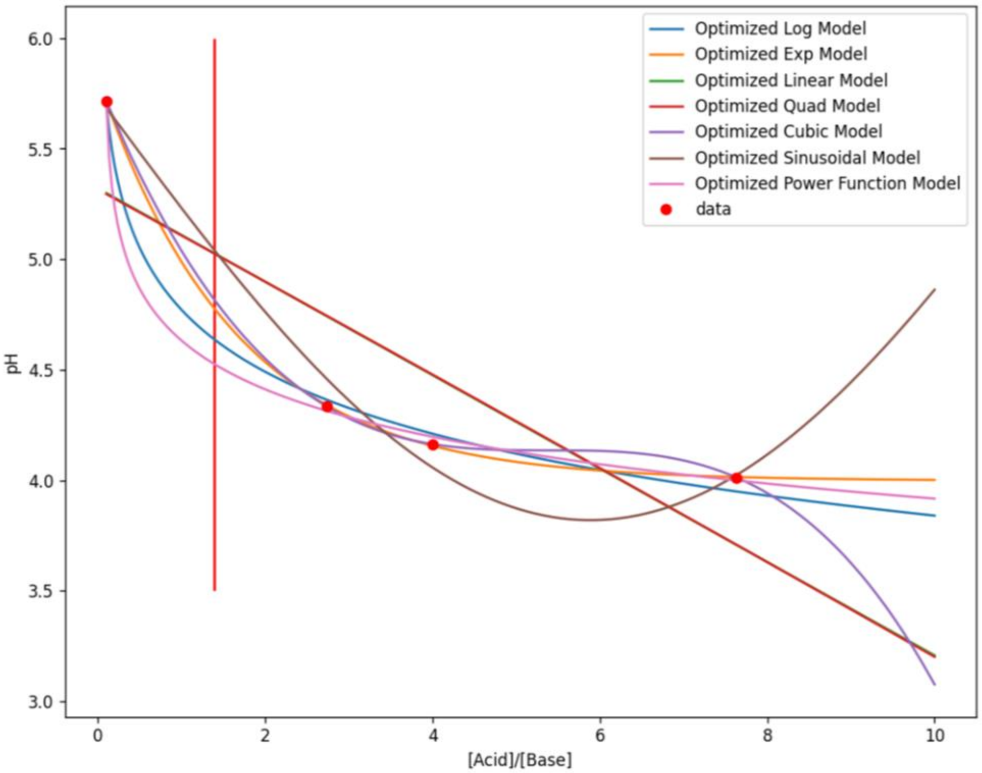

Optimized Log Model
Optimized Exp Model
Optimized Linear Model
Optimized Quad Model
Optimized Cubic Model
Optimized Sinusoidal Model
Optimized Power Function Model
data
pH
0
−20
−40
−60
−80
−100
−120
0
2
4
6
8
10
[Acid]/[Base]



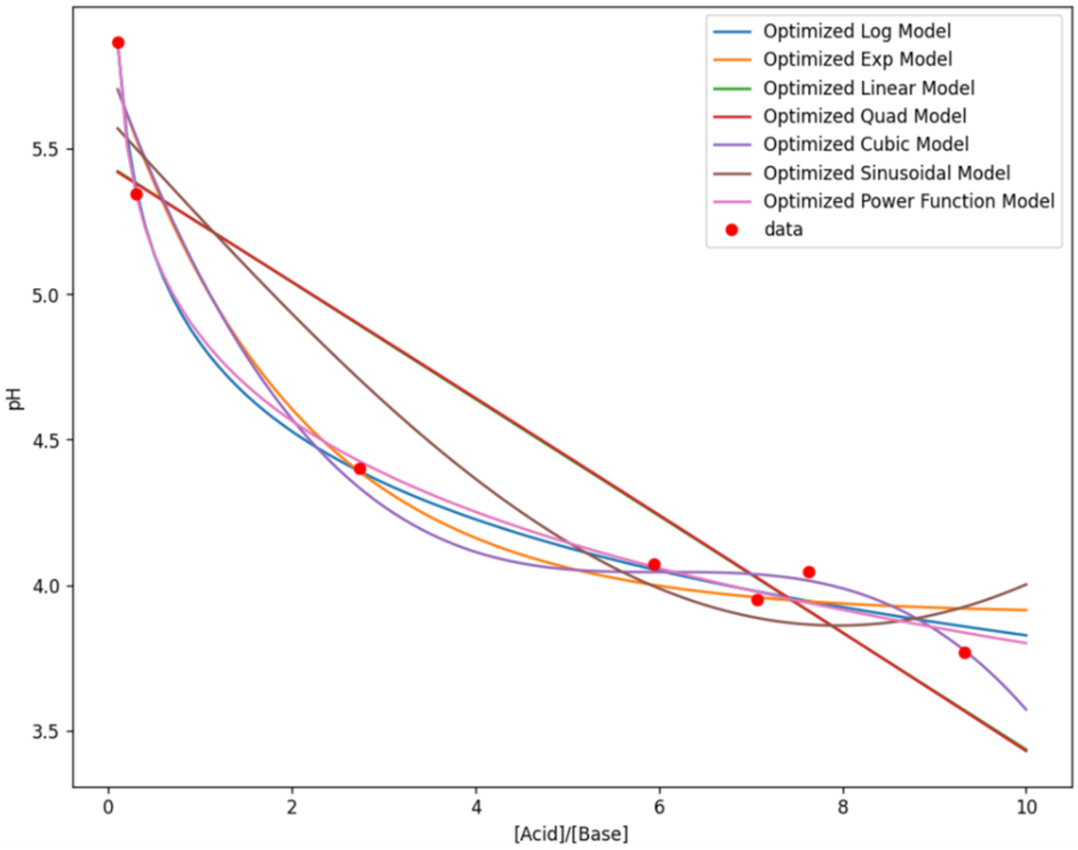
Optimized Log Model
Optimized Exp Model
Optimized Linear Model
Optimized Quad Model
Optimized Cubic Model
Optimized Sinusoidal Model
Optimized Power Function Model
data
pH
[Acid]/[Base]
0
2
4
6
8
10
2
3
4
5



Logarithmic Residuals: [3.93199162e-09 -1.90157090e-08 1.36865332e-08]
Exponential Residuals: [1.02752296e-09 4.78391549e-09 -9.78319159e-09]
Linear Residuals: [-0.22538748 0.34674921 -0.12136261]
Quadratic Residuals: [-0.22826365 0.34965747 -0.12139382]
Cubic Residuals: [0.0000000e+00 -8.8817842e-16 0.0000000e+00]
Sinusoidal Residuals: [0.00000000e+00 8.88178420e-16 2.66453526e-15]
Power Function Residuals: [-1.62216818e-09 5.34302202e-09 -2.93637337e-09]
Standard Deviation of Logarithmic Residuals 1.6798496053153672e-08
Standard Deviation of Exponential Residuals 7.734742674817355e-09
Standard Deviation of Linear Residuals 0.3047650034672648
Standard Deviation of Quadratic Residuals 0.3074907343747346
Standard Deviation of Cubic Residuals 6.280369834735101e-16
Standard Deviation of Sinusoidal Residuals 1.9860273225978185e-15
Standard Deviation of Power Function Residuals 4.461031411149502e-09
Sum of Log Likelihood (Logarithmic Model) -259.10284629868454
Sum of Log Likelihood (Exponential Model) -909.8493580288541
Sum of Log Likelihood (Linear Model) -0.19217208850268835
Sum of Log Likelihood (Quadratic Model) -0.21888844331345877
Sum of Log Likelihood (Cubic Model) -inf
Sum of Log Likelihood (Sin Model) -inf
Sum of Log Likelihood (Power Function Model) -inf
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:65: RuntimeWarning: divide by zero encountered in log
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:66: RuntimeWarning: divide by zero encountered in log
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:67: RuntimeWarning: divide by zero encountered in log
Entropy Acquisition Function
Type here to search



Optimized Log Model
Optimized Exp Model
Optimized Linear Model
Optimized Quad Model
Optimized Cubic Model
Optimized Sinusoidal Model
Optimized Power Function Model
data
pH
[Acid]/[Base]
6.0
5.5
5.0
4.5
4.0
3.5
3.0
0
2
4
6
8
10



Optimized Log Model
Optimized Exp Model
Optimized Linear Model
Optimized Quad Model
Optimized Cubic Model
Optimized Sinusoidal Model
Optimized Power Function Model
data
pH
5.5
5.0
4.5
4.0
3.5
0
2
4
6
8
10
[Acid]/[Base]

Bayesian Inference

Parameter Refinement

Brief Overview - Bayesian Machine Learning

Probabilistic interpretation … quantifying **uncertainty** (how confident are we?)

Bayes Theorem

$$\mathbf{P\,(model \mid data)} = \frac{P(\text{data} \mid \text{model})\, P(\text{model})}{P(\text{data})}$$

Brief Overview - Bayesian Machine Learning

Probabilistic interpretation … quantifying **uncertainty** (how confident are we?)

Bayes Theorem

$$\mathbf{P\,(model \mid data)} = \frac{P(\text{data} \mid \text{model})\,\mathbf{P(model)}}{\cancel{P(\text{data})}}$$

"Posterior"

Our confidence in this model being "correct" given the data (what we want to know)

"Prior"

Our confidence in this model being "correct" before getting data (assumption)

Parameter Refinement - (**Bayesian Inference**)

Prior: Assume model has logarithmic form (pH = A + B*log(C*x))

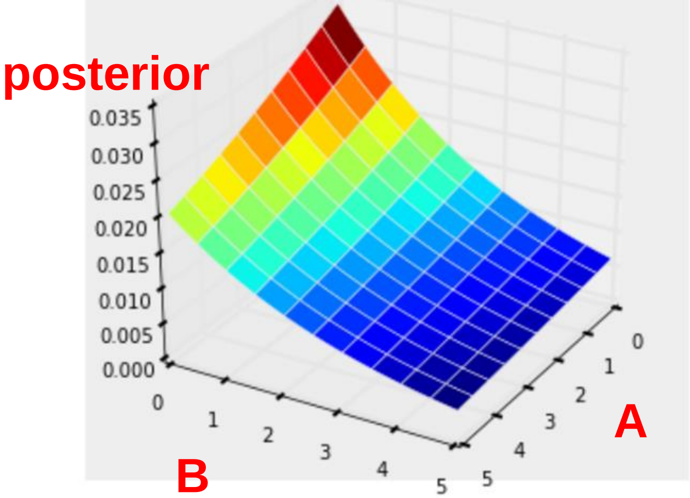
➔ A, B, C are our model parameters

Posterior: Probability of this model and its model parameters given the data

Parameter Refinement - (**Bayesian Inference**)

Assume the model has a certain form

Create parametric model with **model parameters …** (ex: model = **A** + **B** * x)



Problem: How to identify combination of parameters where posterior probability for model is greatest (i.e. best model?)

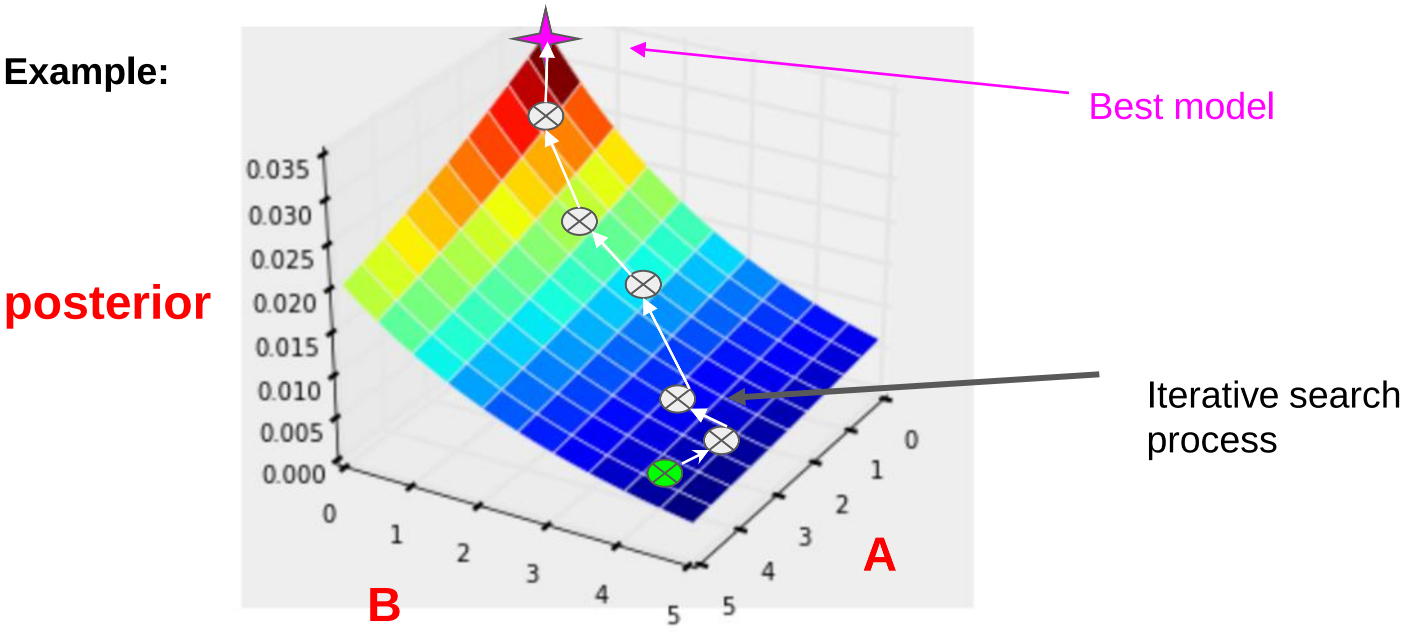
Solution: Sample posterior distribution in parameter space using Markov Chain Monte Carlo (**MCMC**) method

Image Credit: https://nbviewer.jupyter.org/github/CamDavidsonPilon/Probabilistic-Programming-and-Bayesian-Methods-for-Hackers/blob/master/Chapter3_MCMC/Ch3_IntroMCMC_PyMC3.ipynb

Parameter Refinement - (**Bayesian Inference**)

MCMC samples parameter space to find maxima in posterior (best model)

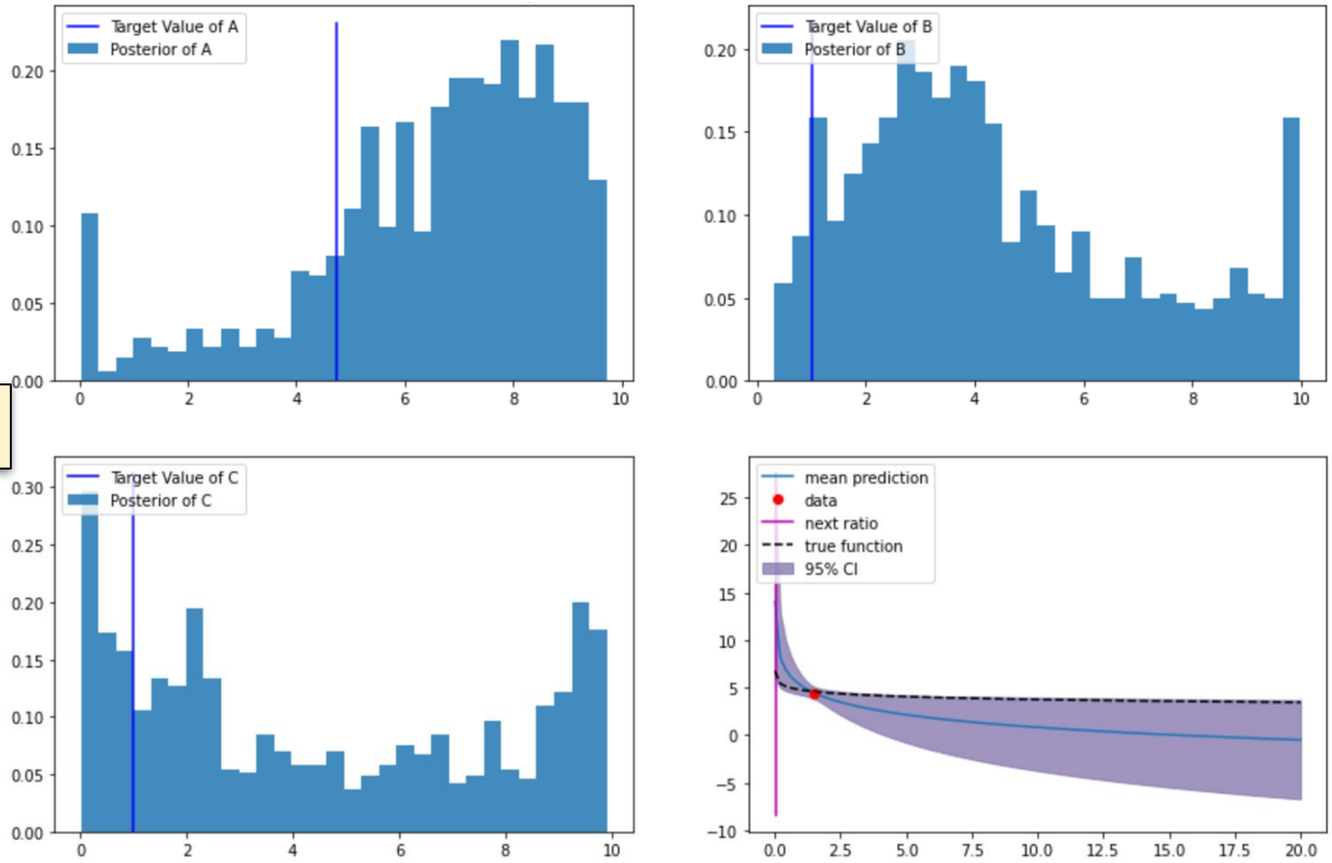
Example:



Autonomous Results - (**Bayesian Inference**)

1 data point

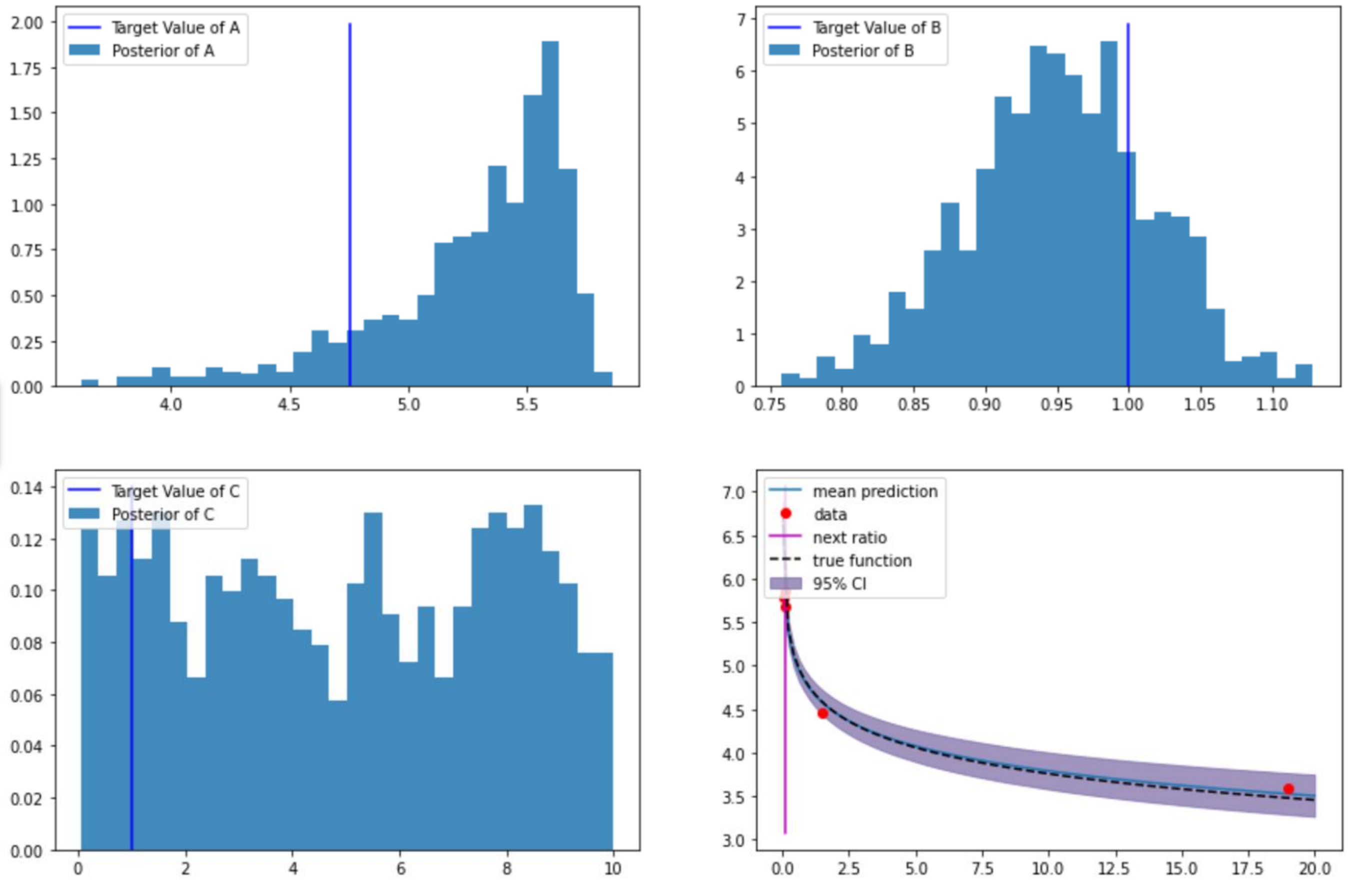
pH = A + B*log(C*x)



Autonomous Results - (**Bayesian Inference**)

5 data points

pH = A + B*log(C*x)

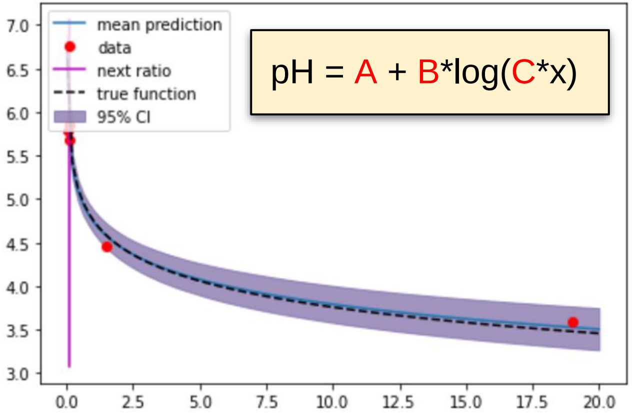


The Next Steps:

For pH Measurement Setup

- ➔ Parameter Refinement

- ➔ Hypothesis Testing
  - ➔ Bayesian methods
  - ➔ Filter between candidate functions

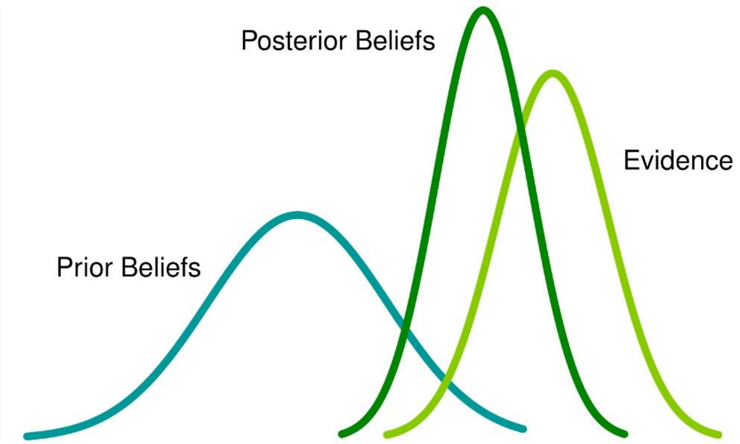


Other Applications (Educational Tool)

- ➔ Camera attachment
  - ◆ Learn color mixing trends

Brief Overview - Bayesian Machine Learning

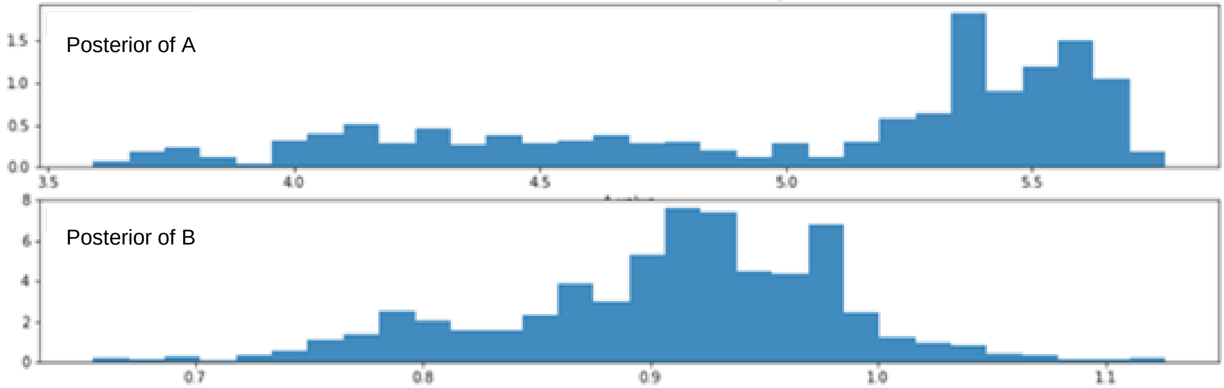
New data alters our prior beliefs → posterior beliefs



Parameter Refinement - (**Bayesian Inference**)

Produces **posterior distributions** for each **model parameter**

Example:



Represent confidence in parameter values

Active learning: [Parameter Refinement] → argmax (variance in model)