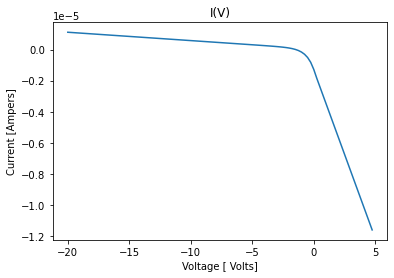
Research Notes, Sources, Information – CIS129 Final Project – Dulcie Quinn

**A bit of background information on why I chose this project:**

This summer I have been working on this project. During the course of my internship at Los Alamos National Lab (LANL), I actually coded this model to predict the plasma parameters, as is proposed in the attached the slides, minus the user interface. Unfortunately I can’t share the actual code due to LANL being tricky with what is allowed to be made public, as technically it’s LANL property and not mine even though I wrote it. However, I did get my research slides approved for unlimited release and all the sources/instructions I used are available to the public, so I thought that since this is a real-world problem that needs to be solved by software, it would be great for this project. So, for this class’s final project, I hashed out the software side of it more specifically in a new deck of slides and with a redesign/proposal of what a user interface might look like in a future, more useable version. My specific summer project was only a proof of concept, so there is still a long way to go before the concept of using ML for this problem can be made into a useable solution. Below are the main sources that I found most useful in completing this project.

**Problem Selection and Justification:**

Space plasma physics is important for scientists to understand. Topics within the subject include the effects of plasma on satellites or space weather from the sun. Important parameters for learning about a plasma species are its density and temperature, which give insight on what type of plasma it is. One way of inferring these parameters is by using Langmuir probes. They are small metal objects used in space and lab experiments that work by being biased to different voltages. Depending on the voltages (positive/negative), the probes attract or repel ions and electrons from the plasma. As the voltage changes, different particles are measured by the probes, which causes the current to change, forming a curve that looks something like this (synthetic data):



Ion saturation region

Retardation region

Linear region

The different regions outlined above are subjective – there is no set boundary for where they start and end; it’s up to the researcher or their software to determine that. Additionally, with real data these curves become much more messy/spiky, making it even more difficult to determine the regions.

The problem is that after measuring, the theory used to infer parameters from the curves (Orbital Motion Limited Theory) doesn’t work in every scenario. It requires specific conditions to be met and is inaccurate if they aren’t. Even when met, it’s a two-step process – first temperature is obtained, then density is obtained using the temperature. A simplified version for density does exist, which doesn’t depend on temperature, but it’s inaccurate for lighter ions such as hydrogen or helium because of the neglected thermal effect, which is significant when the mass of the ion is not.

Machine learning, being something that can be applied as long as you have a dataset and values with correlation, might be a helpful technique to be used in situations where theory doesn’t apply, so in this project I outline a potential technique for it to be used, and how that may want to be set up more effectively, with more sophisticated datasets obtained using kinetic simulations, and trained using much more powerful computers than a laptop. Since the relationship between current/voltage and density/temperature is highly non-linear, a proof of concept is useful to show that machine learning works for this problem, without spending too much computational cost on something that might not work. But since my proof of concept did work, this project’s slideshow shows how it might be further applied.

**Sources (Extended)**

*Existing solution*

<https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2020EA001344>

Resendiz Lira, P. A., & Marchand, R. (2021). Simulation inference of plasma parameters from Langmuir probe measurements. *Earth and Space Science*, *8*, e2020EA001344. https://doi. org/10.1029/2020EA001344

Equations + Theory source. Gives information on how to use OML theory to extract density and temperature from LP measurements.

<https://scikit-learn.org/stable/modules/generated/sklearn.gaussian_process.GaussianProcessRegressor.html>

Documentation for scikit GPR. This was useful in terms of intructions on how to implement/structure my code.

*Existing solution*

<https://agupubs.onlinelibrary.wiley.com/doi/full/10.1002/2016JA022571>

Knudsen, D. J., J. K. Burchill, S. C. Buchert, A. I. Eriksson, R. Gill, J.-E. Wahlund, L. Åhlen, M. Smith, and B. Moffat (2017), Thermal ion imagers and Langmuir probes in the Swarm electric field instruments, J. Geophys. Res. Space Physics, 122, 2655–2673, doi:10.1002/2016JA022571.

More important information on theory. This is the paper that explains how to choose the V\_retardation point by extrapolation.

<https://www.youtube.com/watch?v=iDzaoEwd0N0>

Easy introduction to gaussian process regression (Video)

This is a useful introduction for understanding what the machine is doing without going over the math of it too much, just enough to know whether or not it’s right for this project.

<https://www.geeksforgeeks.org/gaussian-process-regression-gpr/>

Used this article’s “implementation” section to understand how to structure the code, the example was easier for me to understand than only using documentation.