MECENG 249 Project 1 Report

Understanding the gravitational effects on boiling through machine learning

Submitted By:

Joshua Duarte
Pavan M. Reddy

Date: 9-29-2022

Table of Contents

I. Introduction

Boiling heat transfer is a process involving the change in phase from liquid to vapor. Where the viscosity, density, thermal conductivity, and specific heat of the fluid could be used to describe single-phase heat transfer, in boiling many additional properties are required. Boiling is used in a variety of industrial processes and applications, such as refrigeration, power generation, heat exchangers, cooling of high-power electronics components, and cooling of nuclear reactors. In this report, focus is on the aspects of boiling heat transfer in aerospace applications while considering the gravitational effect as the main factor.

In an aerospace system, heat that may have been generated as a byproduct or consequence of a process aboard a vehicle has to be dissipated/dumped out of the system. A common way of heat transfer is done through two-phased loops (transfer of heat through convection) where isothermal efficiencies are achieved.

To illustrate the role of machine learning in heat transfer, an experiment conducted to evaluate the effect of gravitational acceleration on boiling water is considered. The baseline theory behind this experiment is that during the normal earth gravity conditions, the vapors created at the bottom interface escape to the liquid-gas interface letting efficient heat dissipation to occur. However, in microgravity conditions, it is observed that the vapors tend to accumulate at the bottom surface and prevent efficient heat transfer. Now, for increasing the efficiency, a alcohol-water mixture was considered i.e, a water/2-propanol mixture. In this mixture while boiling, the lower vapor pressure alcohol preferentially evaporates at the interface. The lower 2-propanol concentration at the interface near the wall reduces the interfacial concentration of the alcohol there virtually to zero. This, in turn, increases the surface tension close to the contact line at the wall to essentially that of pure water. The resulting gradient in surface tension along the interface produces a Marangoni driven flow of liquid towards the contact line at the wall. This effect is called the Marangoni effect.

The Marangoni effect, which is the mass transfer along an interface between two phases due to a gradient of the surface tension, is an important factor which acts as the main force causing the capillary action during microgravity conditions, as the buoyancy effect is negligible. This mixture provided similar efficiency seen in the boiling water at earth gravity in the micro-g condition. Hence, these data solidify the statement that the gravitational acceleration changes have an effect on boiling processes of water/2-propanol mixture.

So to explore the effects of the gravitational forces and the Marangoni effect on the boiling heat transfer of water/2-propanol mixture, machine learning tools are introduced to the data. A genetic algorithm is used to train the data over a range of gravitational acceleration conditions to better optimize the solution

II. Nomenclature

$c_{p,l}$	liquid specific heat
g	gravitational acceleration, m/s^2
g_{en}	earth-normal gravitational acceleration, 9.8 m/s^2
h_{lv}	latent heat of vaporization, kJ/kg
k_l	liquid thermal conductivity, W/mK
L_h	heated surface characteristic size, m
q''	heat flux, W/cm^2
q''_{en}	heat flux under earth-normal gravity
Pr_l	liquid Prandtl number
T_{sat}	saturation temperature, °C
$T_w - T_{sat}$	wall superheat, °C
$\hat{x}_{1,b}$	bulk mole fraction of 2-propanol in water/2-propanol solution
$\hat{x}_{1,i}$	interface mole fraction of 2-propanol in water/2-propanol solution
α_l	liquid thermal diffusivity
γ	surface tension parameter $=(\sigma_w - \sigma_{mix})/\sigma_{mix}$
μ_l	liquid viscosity
ρ_l	liquid density, kg/m^3
ρ_v	vapor density, kg/m^3
σ	surface tension, N/m

III. Theory

A genetic algorithm is a search heuristic. It is a type of evolutionary algorithm used to perform optimization on constrained or unconstrained data using the principles of natural selection with binary representation and simple operators based on genetic recombination and genetic mutations.

In brief, the algorithm starts with selecting the organisms with the best fitness out of the whole population. These organisms produce offspring that have inherited the characteristics of their respective parents. It is to be noted that if the parents have the best fitness, their offspring are better than the parents and have an increased chance of surviving. This process is run iteratively until a set of best-fit organisms have been produced or the number of generations is completed. These organisms are then used to find the optimal solution to the defined problem.

There are two key steps in using a genetic algorithm to train a model. The first is to frame the problem statement which postulates the function. In this project, the algorithm is used to determine how the heat flux varies with the other parameters of the given data. For this, a postulated equation to solve for the heat flux in terms of superheat, surface tension, pressure, and gravitational acceleration is adopted. For the same, a well-defined initial raw data i.e. the initial population is provided which is the ydata. Once the problem has been defined, we use power law dependence to define the equation. The variables of the equation in each organism are called to be genes. Here the number of genes varies for the different tasks. These genes form a strain called a chromosome which is the solution.

Step 1: Equations adopted to link the parameters:

For a section of the data, when the surface tension and pressure parameters are constant, the equation stated below is used for defining the dependence of the heat flux in terms of the superheat and gravitational acceleration.

$$q'' = n_1 (T_w - T_{sat})^{n_2} g^{n_3} \tag{1}$$

When the whole data is considered while the parameters surface tension and pressure vary, a different equation is used as defined below.

$$q'' = n_1 (T_w - T_{sat})^{n_2} (g + n_4 g_{en} \gamma)^{n_3} P^{n_5} \tag{2}$$

Step 2: Generating the fitness error function which helps to set the survival ratio for the solution

For each equation, we develop an error function which is plugged into the code to perform the survival function where the organisms with error less than the threshold defined survive and the rest are removed. Below are the error functions for the above-postulated relations respectively.

$$f_{err,i} = -ln q''_{data,i} + ln(n_1) + n_2 ln(T_W - T_{sat})_{data,i} + n_3 ln g_{data,i} \tag{3}$$

$$f_{err,i} = -ln q''_{data,i} + ln(n_1) + n_2 ln(T_W - T_{sat})_{data,i} + n_3 ln(g_{data,i} + n_4 g_{en} \gamma_{data,i}) + n_5 ln P_{data,i} \tag{4}$$

Once the algorithm is set up, the data is trained using the initial population and some initial guess for the genes. For the initial run, the guesses can be far from the optimal solution hence the noise generated in the data can be high. Once the understanding of the effect of each gene on the solution is built, the parameters are varied to achieve convergence. There are mainly four parameters that can have an impact on the convergence of the minimum error i.e. NGEN, MFRAC, Perturbation coefficients, and Mutation rate.

NGEN is the variable related to the number of generations. With worse initial guesses, more generations may be required to reach convergence.

MFRAC is the median of the fraction which affects the error threshold for keeping an organism. To prevent from ending at a local error minimum, the value can be raised to potentially find organisms with lower error values to be kept for the next generations.

Perturbation coefficients are the coefficients in the initial population array which define the range of initial guesses over the population. With worse initial guesses the algorithm needs to be started with a lower range of perturbation constants to search for solutions over a wider domain.

The mutation rate is the parameter that controls the level of diversity in the offspring formed. This allows the algorithm to sample more of the parameter space and avoid landing up in a local error minimum. With worse initial guesses, the mutation rate is kept high to search over a larger solution domain. However, it is helpful to decrease the mutation rate as the solution is honed to fine-tune the final constants.

IV. Task One

a)

Within this task, we were able to uncover if our Python environment and respective libraries were functional for use throughout the remainder of the tasks. To do so, CodeP1.1 was provided and inserted within the initial cells of the Jupyter Notebook. The output of data that was collected in the array variable *ydata* was compared and verified against the raw data given in the project file.

```
  ydata = [[44.1, 32.5, 0.098, 1.79, 5.5], [47.4, 33.2, 0.098, 1.79, 5.5], [49.4, 34.2, 0.098, 1.79, 5.5], [59.2,
34.8, 0.098, 1.79, 5.5], [67.8, 36.3, 0.098, 1.79, 5.5], [73.6, 37.3, 0.098, 1.79, 5.5], [76.3, 37.8, 0.098, 1.79,
5.5], [85.3, 39.2, 0.098, 1.79, 5.5], [96.5, 39.3, 0.098, 1.79, 5.5], [111.0, 42.3, 0.098, 1.79, 5.5], [124.0, 43.5,
0.098, 1.79, 5.5], [136.2, 45.4, 0.098, 1.79, 5.5], [143.5, 46.7, 0.098, 1.79, 5.5], [154.6, 47.9, 0.098, 1.79,
5.5], [163.1, 48.6, 0.098, 1.79, 5.5], [172.8, 50.9, 0.098, 1.79, 5.5], [184.2, 51.7, 0.098, 1.79, 5.5], [203.7,
56.4, 0.098, 1.79, 5.5], [36.7, 30.2, 9.8, 1.79, 5.5], [55.1, 34.1, 9.8, 1.79, 5.5], [67.5, 35.3, 9.8, 1.79, 5.5],
[78.0, 37.8, 9.8, 1.79, 5.5], [92.0, 38.1, 9.8, 1.79, 5.5], [120.0, 44.1, 9.8, 1.79, 5.5], [134.3, 46.9, 9.8, 1.79,
5.5], [150.3, 48.5, 9.8, 1.79, 5.5], [167.0, 49.2, 9.8, 1.79, 5.5], [184.0, 52.7, 9.8, 1.79, 5.5], [196.5, 53.1, 9.8,
1.79, 5.5], [42.4, 29.7, 19.6, 1.79, 5.5], [48.7, 31.0, 19.6, 1.79, 5.5], [54.5, 31.2, 19.6, 1.79, 5.5], [70.8, 32.4,
19.6, 1.79, 5.5], [73.7, 31.4, 19.6, 1.79, 5.5], [81.8, 32.5, 19.6, 1.79, 5.5], [91.9, 36.3, 19.6, 1.79, 5.5], [103.9,
36.3, 19.6, 1.79, 5.5], [119.1, 37.2, 19.6, 1.79, 5.5], [133.7, 38.4, 19.6, 1.79, 5.5], [139.9, 39.7, 19.6, 1.79,
5.5], [148.3, 40.9, 19.6, 1.79, 5.5], [157.0, 41.6, 19.6, 1.79, 5.5], [169.1, 43.9, 19.6, 1.79, 5.5], [179.2, 45.0,
19.6, 1.79, 5.5], [205.0, 47.9, 19.6, 1.79, 5.5]]
```

Output 1 : CodeP1.1 ydata

b)

With the array of data obtained within Task 2a, two log-log plots of heat flux versus wall superheat were created with the gravitational accelerations being 0.098 and 9.8 m/s^2 respectively. This provided a visual representation of the variation of heat flux with different gravity and superheat. The plots were created using libraries *NumPy* and *matplotlib* and can be found below.

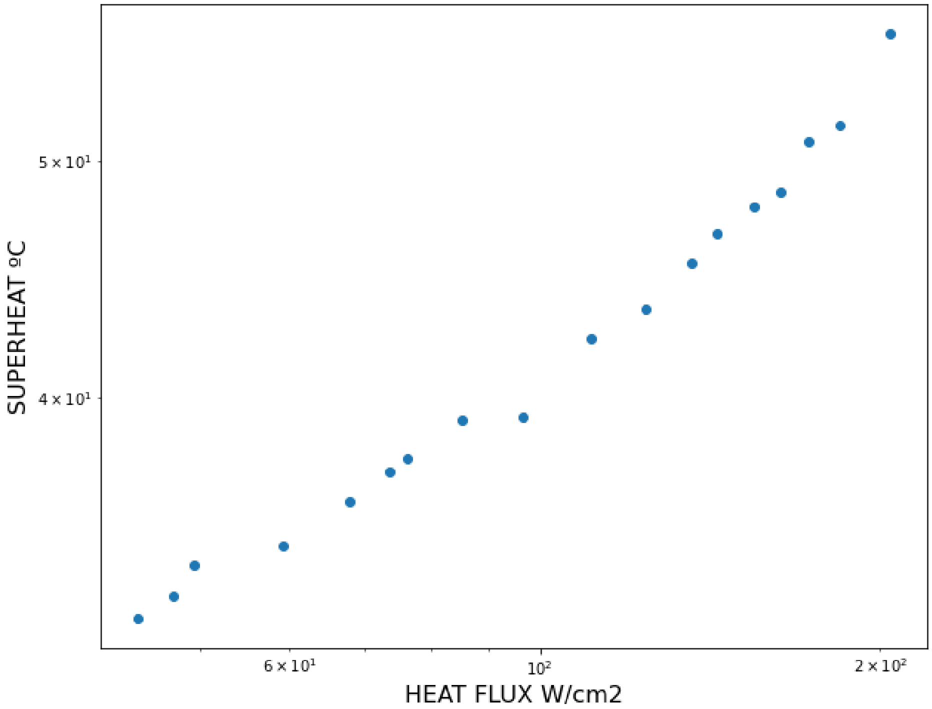

Figure 1: Heat Flux vs. Wall Superheat where g = 0.0098 m/s^2

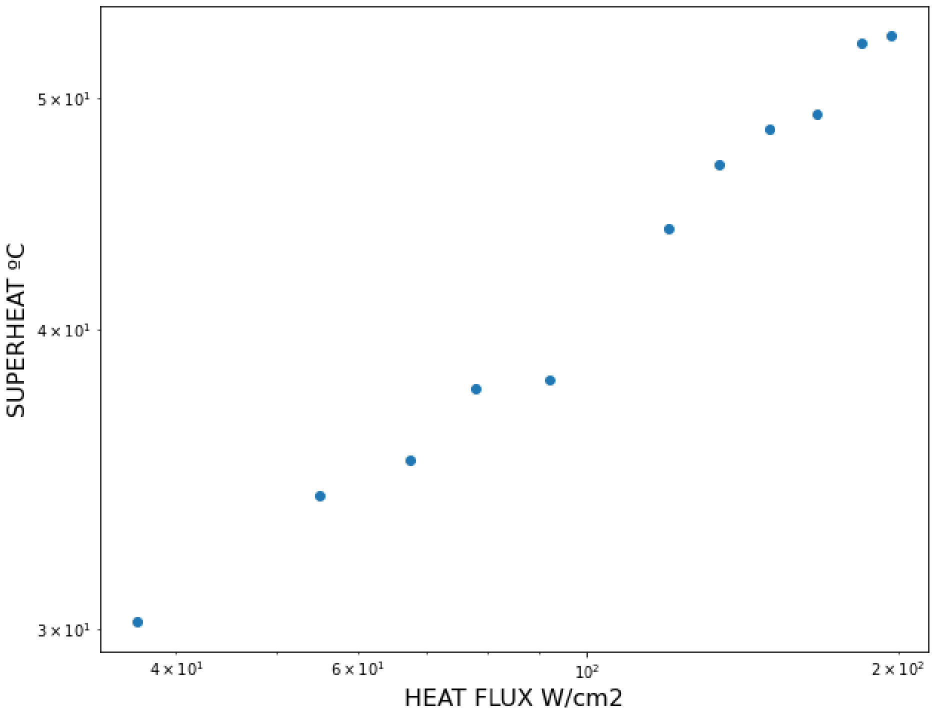

Figure 2: Heat Flux vs. Wall Superheat where g = 9.8 m/s^2

V. Task Two

This task developed familiarity with the elements that comprise the genetic algorithm that explores the interplay of gravity and Marangoni effects during the boiling process for a variety of gravitational acceleration conditions. Manipulation of the genetic algorithm was conducted to study the convergence of the algorithm. Convergence was indicated when the algorithm produced a minimum error, $[[F_{err}]_{mean}/N_D]_{min}$, less than 0.04.

To manipulate the genetic algorithm, four different initial inputs for genes 1-3 were selected to visualize the effects on the genetic algorithm. Furthermore, the number of generations (NGEN), Fraction of median threshold (MFRAC), and the mutation rate (MR) were adjusted until the algorithm produced convergence. A plot was also created to provide a greater visualization of the fluctuations of the gene values and average error for the population throughout the generation. A second plot was created to show the measured heat flux vs the predicted heat flux using the gene constants that resulted from the genetic algorithm. Below lies the initial case of values provided for the script along with the four other trials conducted with their respective minimum errors.

Trial	Gene n1i	Gene n2i	Gene n3i	NGEN	MFRAC	MR	Min error
Base	0.00027	4.0	0.063	6000	0.5	0.09	0.031

1	0.029	4.6	0.03	6000	0.8	0.25	0.030
2	0.0035	2.7721	0.02836	6000	0.4	0.06	0.0293
3	0.029	5.6	0.0788	18000	0.8	0.07	0.0307
4	1.1	1.2	0.35	18000	0.9	0.4	0.038

Table 1: Task Two Trials and Respective Inputs

The base trial consisted of initial values provided within the project instructions which served as a baseline of the genetic algorithm. These values ultimately led to convergence with a minimum error of around 0.031 and an RMSE deviation of 31.040 which lies close to the ± 10% error range as the scale of the plot ranges from 50 to 200. Within the plot created to show the fluctuations of the constants and errors throughout the generation, there was little noise within the constants. However, noise progressively increased within the error as the generation number increased.

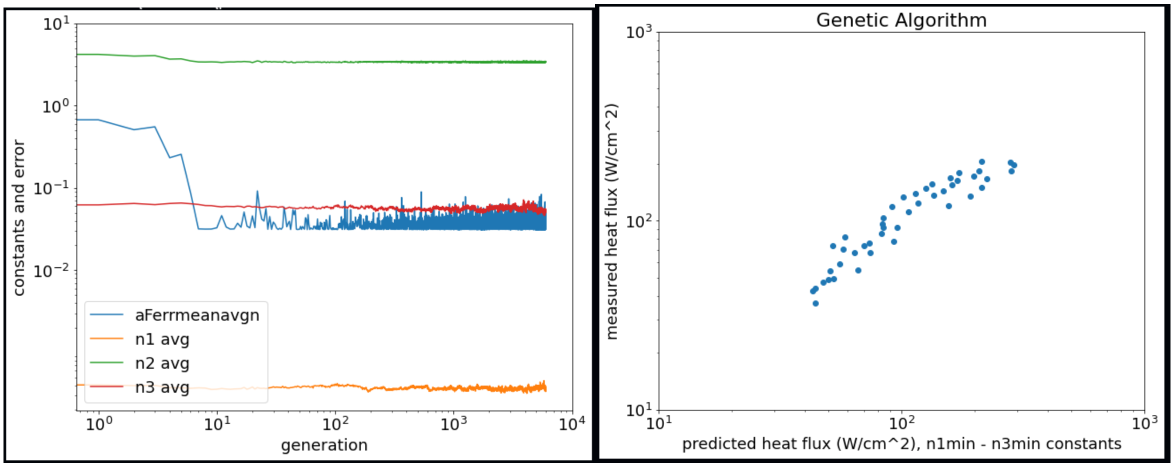

Figure 3: Base Trial Constants and Error and Measured Heat Flux vs Predicted Heat Flux

Within trial 1, the values of the genes (n1i, m2i, and n3i) were altered from the provided initial gene inputs. Without manipulation of the fractional median threshold, mutation rate, or the number of generations, convergence was not achieved. Both MFRAC and MR were then raised to analyze the effect they had on the genes throughout the algorithm. What was found was that with a heightened mutation rate, there is significantly more noise within the plot of the data. This noise would arise as the generations progressed with the most noise present at the end of the generations. The fractional median threshold was also raised to accept more values and prevented the algorithm from being susceptible to a local minimum. This raised fractional median threshold is also seen to contribute to the noise present within the data. The heightened MFRAC and MR led to convergence as a greater range of data could be analyzed even though the initial values used were dramatically altered. The resulting aFerravgMin and RMSE were determined to be around 0.030 and 21.563 respectively.

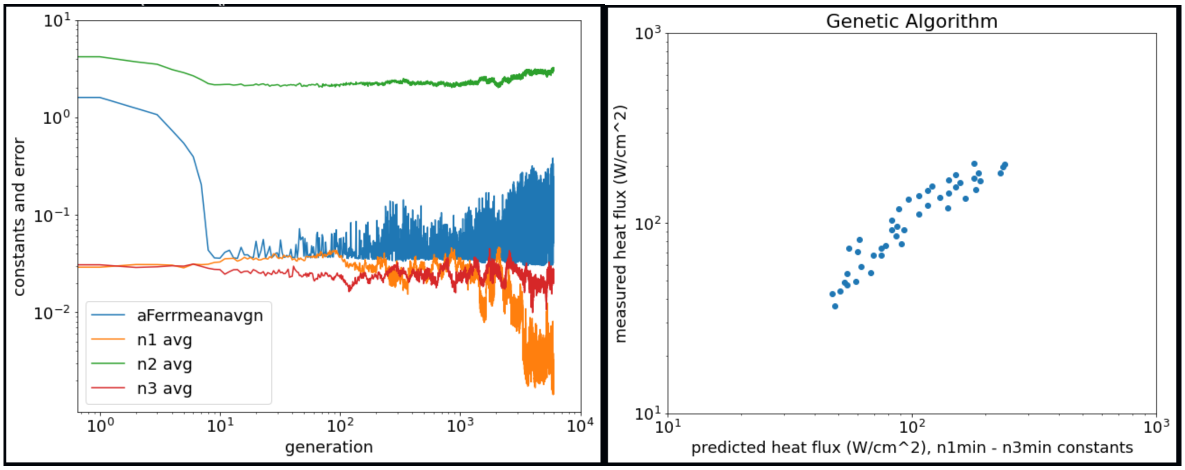

Figure 4: Trial 1 Constants and Error and Measured Heat Flux vs Predicted Heat Flux

For trial 2, the gene values (n1i, n2i, and n31) were selected as the previous trials determined gene values associated with minimum. With this data that already converged, the fractional median threshold and the mutation rate were manipulated in an attempt to narrow in on values that would bring the error present down. The selected values did just that as the outcome presented minimum error and RMSE values of around 0.0293 and 19.697 respectively.

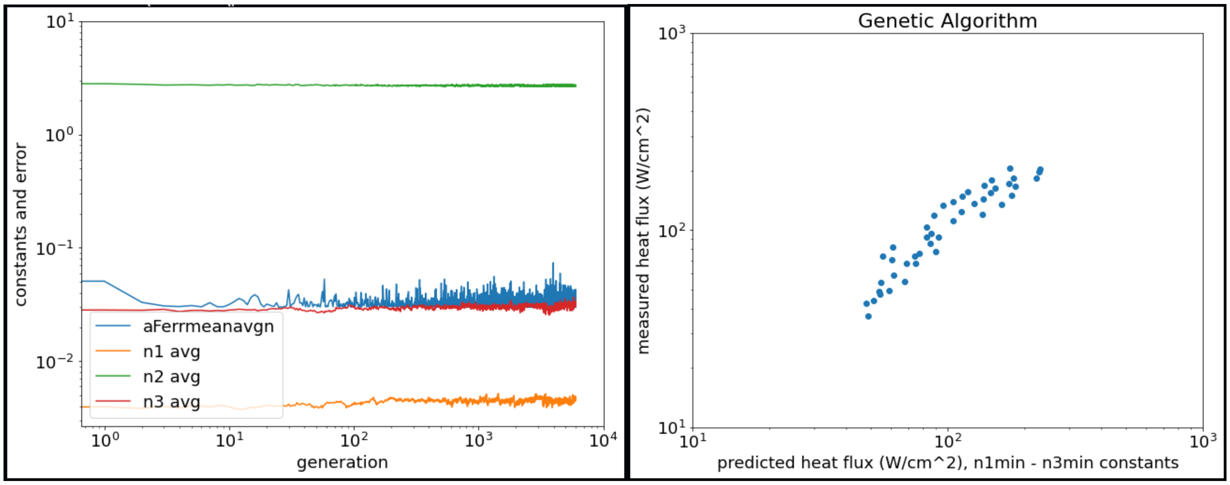

Figure 5: Trial 2 Constants and Error and Measured Heat Flux vs Predicted Heat Flux

In Trial 3, the initial gene values were again dramatically changed to determine the best combination of the MFRAC and MR that would result in convergence. This time, however, the number of generations was tripled to 18000. Initially, the gene values and number of generations with the baseline MFRAC and MR were insufficient as they did not result in convergence. This led to the increase of MFRAC to 0.8 and left MR at the provided 0.09 value. This combination brought the convergence of the algorithm, but significant noise was present within the constants. To reduce this, the MR was lowered to 0.07 which drastically reduced the noise altogether throughout the 18000 generations. The recorded minimum error was around 0.0307 with an RMSE value 18.978.

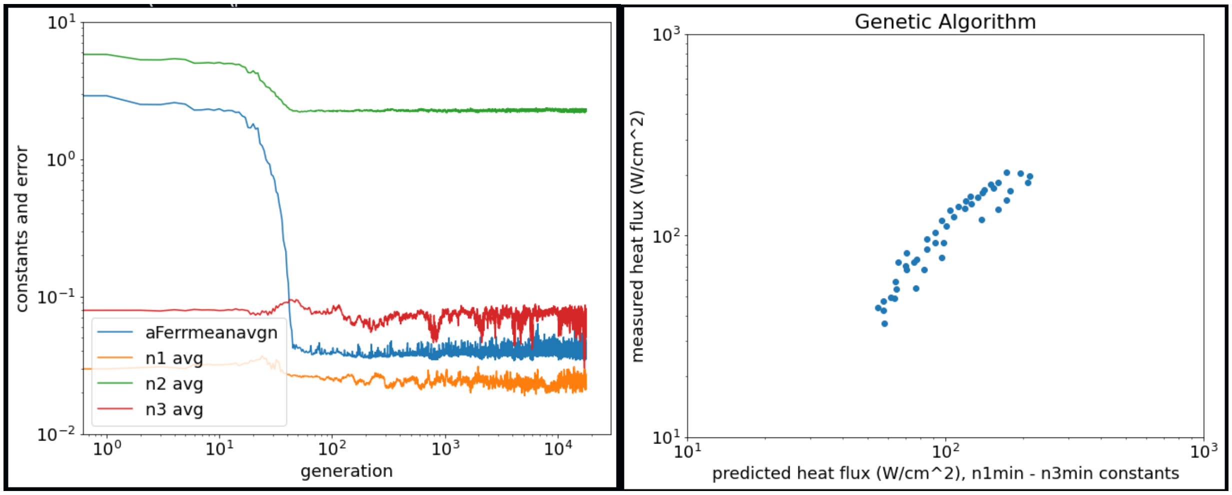

Figure 6: Trial 3 Constants and Error and Measured Heat Flux vs Predicted Heat Flux

Trial four saw gene values that were the most different from any that were previously selected along with the number of generations being 18000. With the insight gained from the previous trials, it was understood that both the MFRAC and MR would have to be high values to allow the algorithm to traverse through a wider range of values. This led to the MFRAC and MR being 0.9 and 0.4 respectively. These high values led to convergence, but an extreme amount of noise in the constants and minimum error averages throughout the generations. These manipulations along with the initial gene selection resulted in a minimum error and RMSE of around 0.038 and 19.0055 respectively.

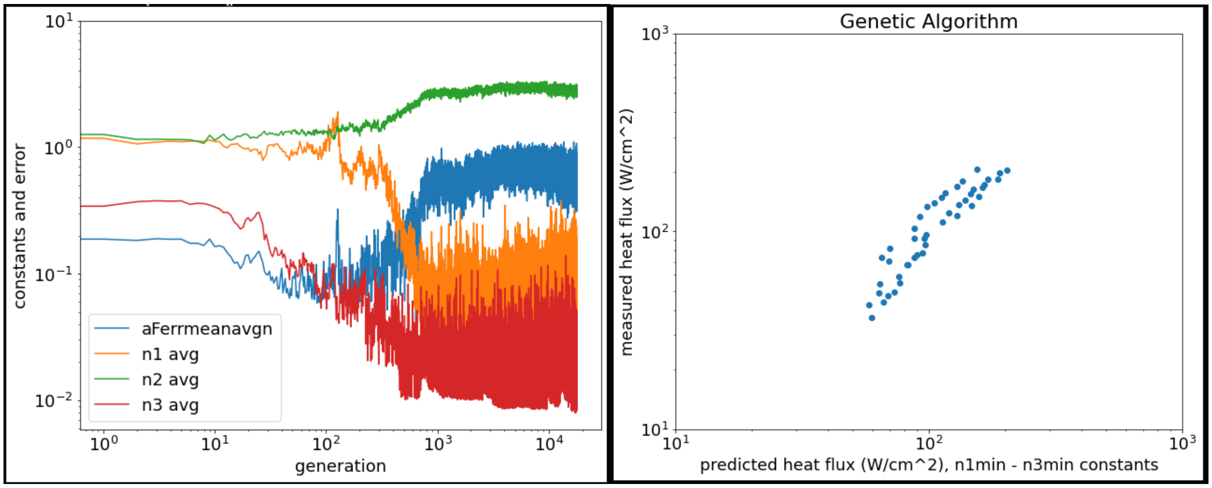

Figure 7: Trial 4 Constants and Error and Measured Heat Flux vs Predicted Heat Flux

In order to go through the trials in a timely manner and improve the readability of the scripts, the initially provided script was adjusted. First, the mutation rate and perturbation constants were changed from hard values to variables that could be adjusted as necessary. Every variable that could potentially be changed to achieve convergence of the trial was removed from the main script cell and placed above it. These variables included the initial gene values, the number of generations, the fraction of median threshold, mutation rate, and the perturbation constants. To increase readability, the commented code that would be used in later tasks was removed. Each trial saw a change in values from the baseline provided. For trial one, the gene values, the MFRAC, and the MR were changed. For trial two, the gene values, the MFRAC, and the MR were changed. For trial three, the gene values, the NGEN, the MFRAC, and the MR were changed. Lastly, trial three saw the gene values, the NGEN, the MFRAC, and the MR change.

VI. Task Three

With the previous task creating some familiarity with genetic algorithms and the manipulation required to ensure conversion, task three introduces more data and two additional gene values to create a five-constant model. This new model includes varying pressures and the surface tension parameter γ. To fit this new model and data set, the provided script was altered to include commented code, and the number of data vectors, ND, and the total number of DNA strands, NS, were changed from 45 to 77. In doing so, other aspects of the script that were meant to accommodate only three constants had to be adjusted to fit the five. This was done by using the equation below and the guide that was provided within the project instructions.

$$f_{err,i} = -ln q''_{data,i} + ln(n_1) + n_2 ln(T_w - T_{sat})_{data,i} + n_3 ln(g_{data,i} + n_4 g_{en} \gamma_{data,i}) + n_5 ln P_{data,i} \qquad (5)$$

The result of the adjustments was a minimum error of around 0.0285. This minimum error ensured that the guesses created the best fit.

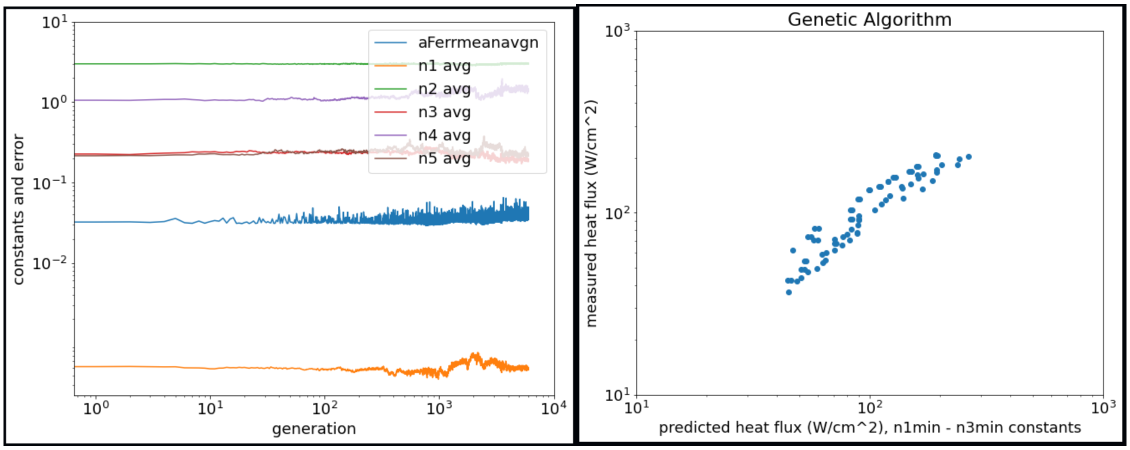

Figure 8: Task 3 Constants and Error and Measured Heat Flux vs Predicted Heat Flux

With initial guesses that resulted in the best fit, a surface plot of $q''/ (T_w - T_{sat})^{n2}$ versus g and γ for $1.0 < g < 20$ m/s2 and $0.0 < \gamma < 2.0$ at a pressure of 10 kPa was created. This creates a greater visualization of the interactions between the heat flux and the varying surface tension parameter and gravity at a set pressure.

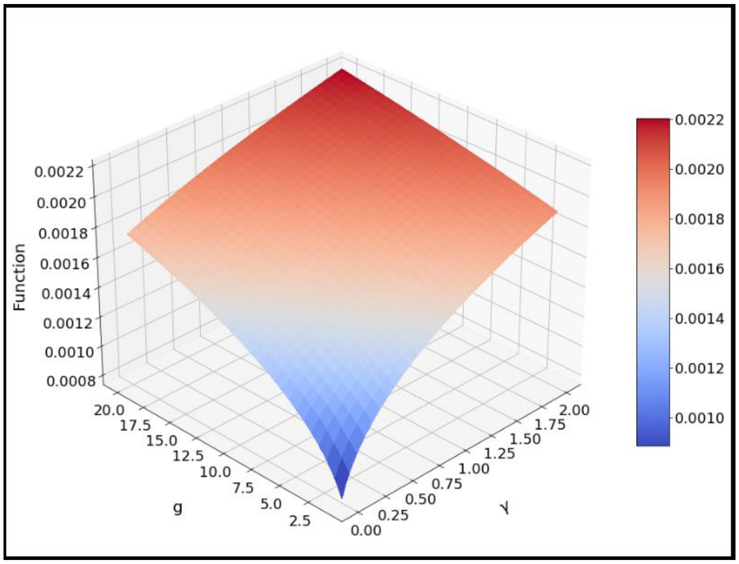

Figure 9: Surface Plot of Function = $q''/(T_w - T_{sat})^{n2}$ *vs.* g *and* γ

VII. Task Four

Using dimensionless equations and data is a typical solution to generalize and reduce the complexity of problems. Through the well-known Rohsenow correlation for nucleate boiling, the data from Task Three was converted to non-dimensional parameters. The equations for conversion are found below:

$$Q_s = 8\frac{q''}{\mu_l h_{lv}}\sqrt{\frac{\sigma}{g_{en}(\rho_l - \rho_v)}} \tag{6}$$

$$J_{as} = 100\frac{c_p(T_w - T_{sat})}{h_{lv}} \tag{7}$$

$$\mathrm{Pr}_l \tag{8}$$

$$\frac{g}{g_{en}} \tag{9}$$

$$\gamma = \frac{\sigma_w - \sigma_{mix}}{\sigma_{mix}} \tag{10}$$

Along with the equations, properties were provided for the three varying pressures seen within the data (5.5 kPa, 7.0 kPa, and 9.5 kPa).

	P = 5.5 kPa	P = 7.0 kPa	P = 9.5 kPa
T_{sat} (°C)	34.9	38.0	45.0
c_{pl} (kJ/kg°C)	4.18	4.18	4.18
h_{lv} (kJ/kg°C) 2	2418	2406	2394
μ_l (Ns/m^2)	7.19x10^{-4}	6.53x10^{-4}	5.96x10^{-4}
Pr_l	4.83	4.54	3.91
ρ_l (kg/m^3)	994	993	990
ρ_v (kg/m^3)	0.0397	0.0476	0.182
σ (N/m)	0.0706	0.0692	0.0688

Table 2: Low Pressure Water Saturation Properties

With the equations for non-dimensional conversion and the properties for the values at their respective pressures, a for loop with conditions set to recognize the pressures, the data was converted. With the dimensionless parameters set, an error function, $f_{err,i}$, was created to fit within the genetic algorithm.

$$Q_s = n_1 Ja_s^{n_2} \left(\frac{g}{g_{en}} + n_3\gamma\right)^{n_4} Pr_l^{-n_5} \tag{11}$$

Equation (11) stated above was altered using the following steps to obtain the error function.
Taking natural log on both sides:

$$ln(Q_s)_{data,i} = ln(n_1)_{data,i} + n_2 ln(Ja_s)_{data,i} + n_4 ln(\frac{g}{g_{en}} + n_3\gamma)_{data,i} - n_5 ln(Pr_l)_{data,i} \tag{12}$$

Based on this equation, an error function $f_{err,i}$ for a given data point *i* can be defined as:

$$f_{err,i} = -ln(Q_s)_{data,i} + ln(n_1)_{data,i} + n_2 ln(Ja_s)_{data,i} + n_4 ln(\frac{g}{g_{en}} + n_3\gamma)_{data,i} - n_5 ln(Pr_l)_{data,i} \tag{13}$$

With a total of N_D data points in the data set to be analyzed, the total error function *Ferr* is the sum of the fractional absolute value of the error for each data point in the population, N_D:

$$F_{err} = \sum_{i=1}^{N_D} \frac{|f_{err,i}|}{|ln(Q_s)_{data,i}|} = \sum_{i=1}^{N_D} \frac{\left|-ln(Q_s)_{data,i} + ln(n_1)_{data,i} + n_2 ln(Ja_s)_{data,i} + n_4 ln(\frac{g}{g_{en}} + n_3\gamma)_{data,i} - n_5 ln(Pr_l)_{data,i}\right|}{|ln(Q_s)_{data,i}|} \tag{14}$$

With the equations defined, they were translated within the code at the necessary lines to incorporate the changes done for the dimensionless parameters as follows:

```
Ferravgn[i] = -1.*lydata[i][0] + math.log(n1avg[k]) + n2avg[k]*lydata[i][1]
                     + n4avg[k]*math.log(ydata[i][2] + n3avg[k]*ydata[i][3])
                    - n5avg[k]*math.log(ydata[i][4])                                    (15)
```

In altering the provided script with the non-dimensional conversion of the data and replacing the error and heat flux equations derived above, the non-dimensional data was obtained as output. This output can be found in Appendix 2.2.

VIII. Task Five

a)

Using the script seen within Task Three and adding the changes discussed in Task Four, gene constants n_1 through n_5 were to be determined to create a best-fit data set. To do this, the understanding of the number of generations, the fraction of median threshold, and the mutation rate were vital to determine suitable values. Initially, the values used were ones that were provided within Task Three. These values resulted in a minimum error of around 0.1, which was unacceptable. To combat this, the number of generations, the fraction of median threshold, and the mutation rate were raised. This led to new minimum gene values that were then substituted in for the previous initial gene values. This process was repeated until there was a negligible difference between the new minimum error value and the previous generation's minimum error.

The minimum error did not achieve a convergence level (less than 0.04) until the mutation rate was lowered to 0.15 with a fraction of median threshold value of 0.9 and a number of generations being 20,000. The gene values that were associated with this minimum error were then substituted for the previously used values. The generations were then lowered to 6,000 with a lower mutation rate of 0.06 and maintaining the median threshold value of 0.9. These settings with the final genetic values lead to a minimum error that is consistently below 0.04 and around 0.385 with an RMSE value of 2.234..

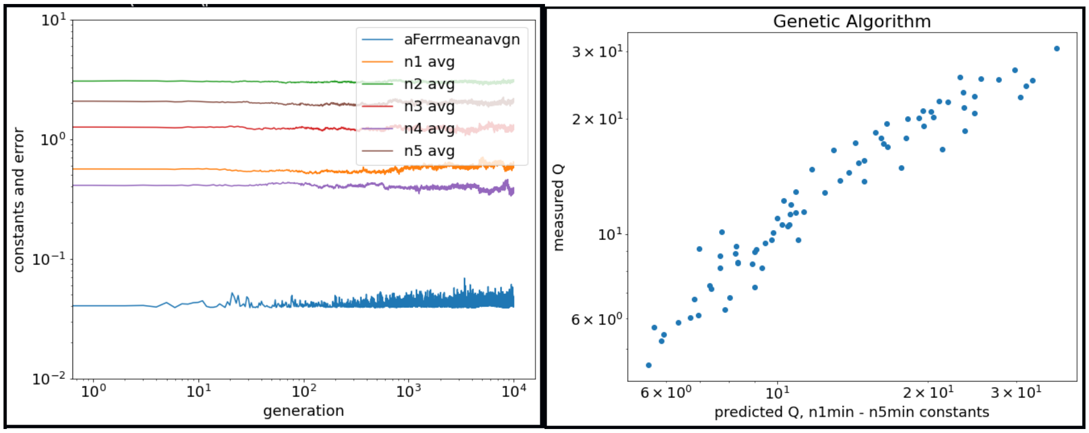

Figure 10: Task 5.a Constants and Error and Measured Q vs Predicted Q

b)

To generate the surface plot of the function varying over a range of g/g_{en} and γ at a pressure of 10kPa, the equation (11) was rearranged to obtain the below-stated form:

$$\frac{Q_s Pr_l^{-n_5}}{J a_s^{n_2}} = n_1\left(\frac{g}{g_{en}} + n_3\gamma\right)^{n_4} \tag{16}$$

Next we used the data of constant values from the best fit derived in task 5(a) to determine and plot the results.

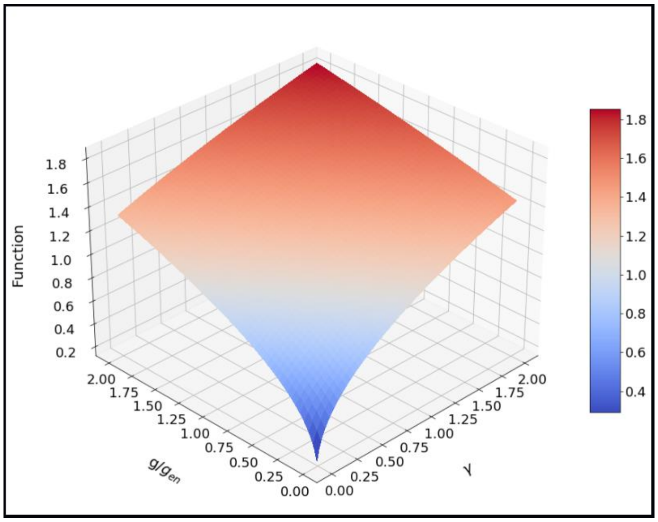

Figure 11: Surface Plot of Function = $Q_s Pr_l^{n5}/Ja_s^{n2}$ vs. g/g_{en} and γ

c)

With the genetic algorithm trained off of the 75 data points, this task aims to test and validate the training through a data set containing 15 data points. To do this, the script's number of data vectors in the array, ND, and the total number of DNA strands, NS, were changed from 75 to 15. The provided validation data was then inserted into the ydata array variable. Then using the low-pressure saturation properties given in task 4 the validation data was converted to non-dimensionless data to determine the $Q_{s,exp}$ using equation (11). In determining the RMSE deviation value and plotting the values, the predicted heat flux data was converted back to dimensional values. The created plot provided a visual representation of the variation of the measured vs predicted heat flux data. The RMSE deviation value was determined to be 14.70872.

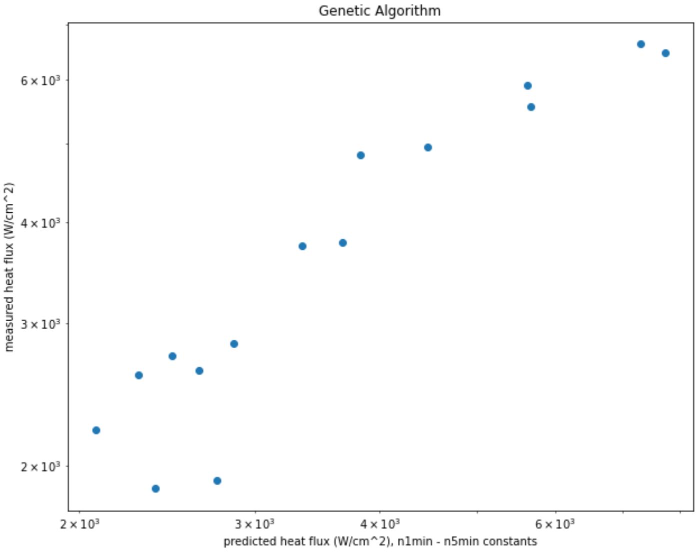

Figure 12: Task 5.c plot between Measured q" vs Predicted q"

IX. Conclusion

The use of genetic algorithms allowed for the simplification of a complex problem. Throughout the tasks above, a greater understanding of genetic algorithms and manipulation of said algorithms was established. More specifically, the mutation rate, number of generations, and the fraction of median threshold were better understood. It was determined that the combination of these values can allow for the algorithm to search across a wider or smaller range of potential solutions. Adjusting these values specifically, the number of generations could directly lead to the increase or decrease of operation time. Within these tasks, a greater understanding of raw data analysis and dimensionless data was developed.

Use of raw data analysis requires high computational power due to the higher number of variables present in the system to determine, and more difficult to visualize the interplay of said variables and their effect on the overall system. The raw data analysis also typically leads to a larger scale of data that can be more difficult to comprehend. On the other hand, in dimensionless analysis the number of variables present is minimized. This helps to achieve an optimal solution with minimal resources. Dimensionless analysis can also be scaled to bigger or smaller systems. This allows for validation of the results using a smaller set of untrained data

The genetic algorithm was successfully trained to optimize the solution accurately. The plots that define the variation between the measured and predicted heat flux data explain that gravitational acceleration plays an important role on the heat dissipation rate. Also, during lower gravity conditions marangoni effect acts as a supplement for the absence of the buoyancy effect present in higher gravitational acceleration conditions.

In completing the tasks, the potential for machine learning and genetic algorithms was made visible. It also made aware of the necessity of data, initial guesses, and other parameters affecting the algorithm. Without proper alignment, a genetic algorithm can produce a solution that does not meet the requirements of the problem.

X. Work Distribution

Throughout the project timeline, the work was divided by task, but was ultimately worked on by both teammates as needed. Using GitHub, and living close in proximity, the team was able to easily collaborate. The report was also worked on by task and revised by one another.

XI. Appendices

1. Figures

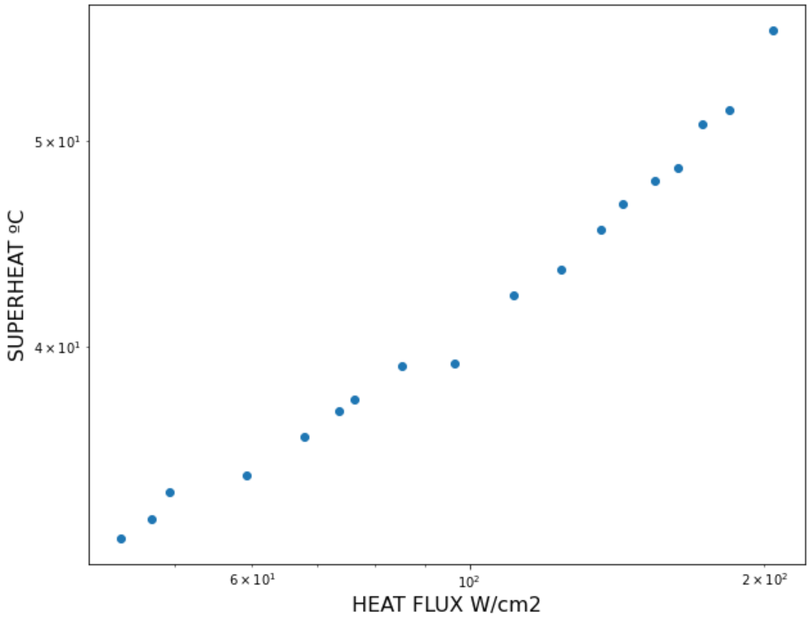

Figure 1: Heat Flux vs. Wall Superheat where g = 0.0098 m/s^2

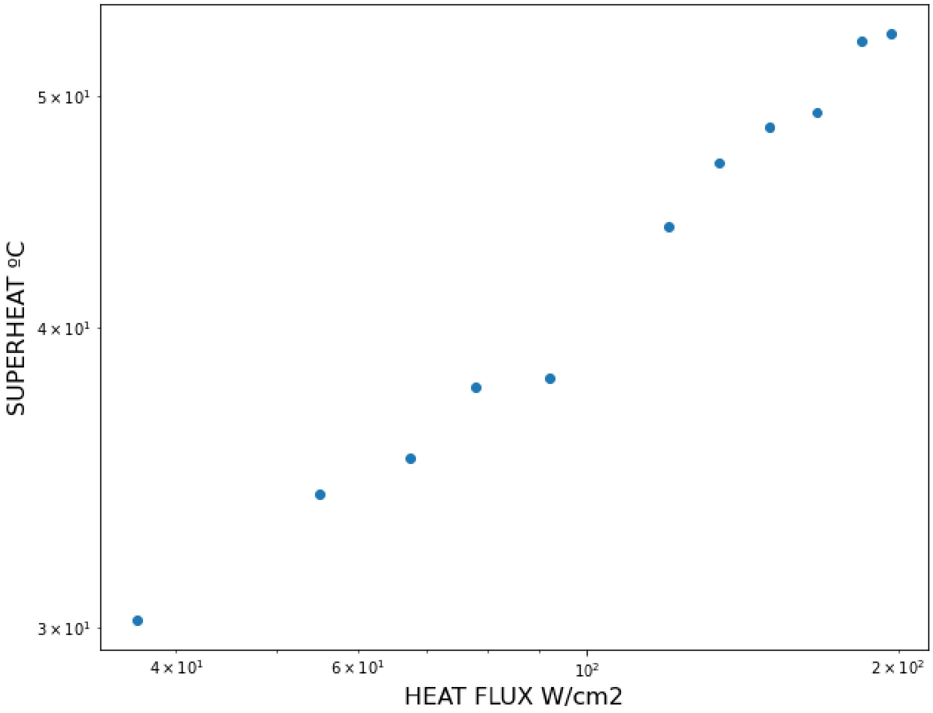

Figure 2: Heat Flux vs. Wall Superheat where g = 9.8 m/s^2

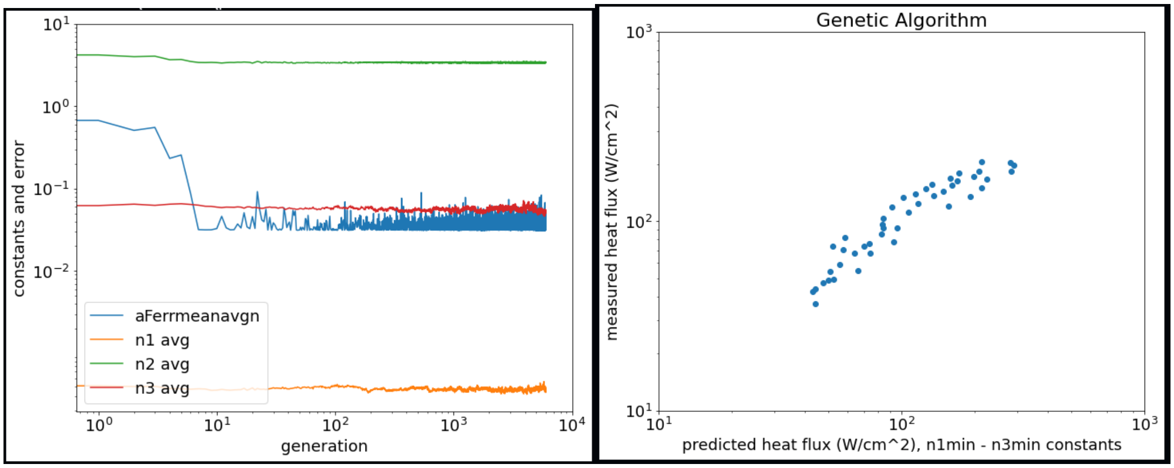

Figure 3: Base Trial Constants and Error and Measured Heat Flux vs Predicted Heat Flux

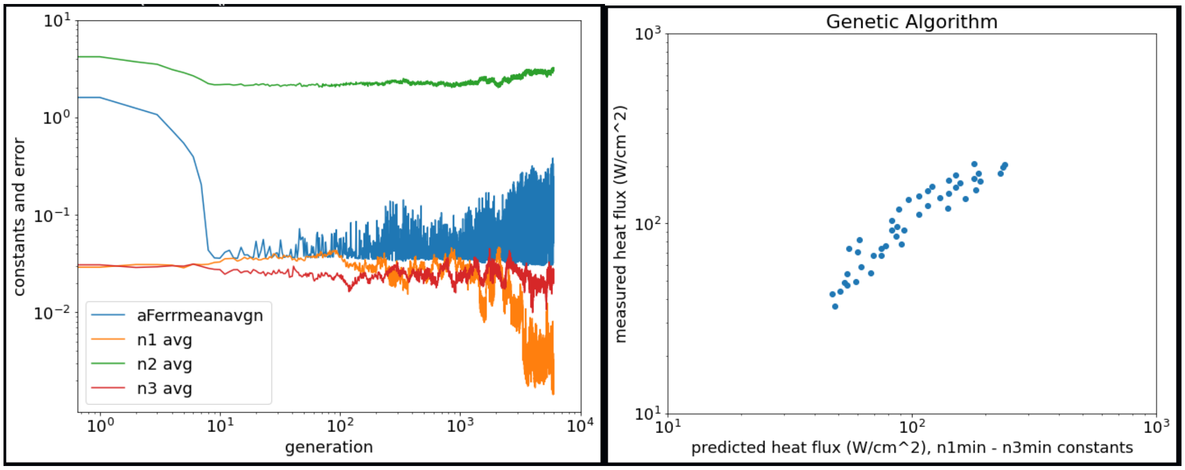

Figure 4: Trial 1 Constants and Error and Measured Heat Flux vs Predicted Heat Flux

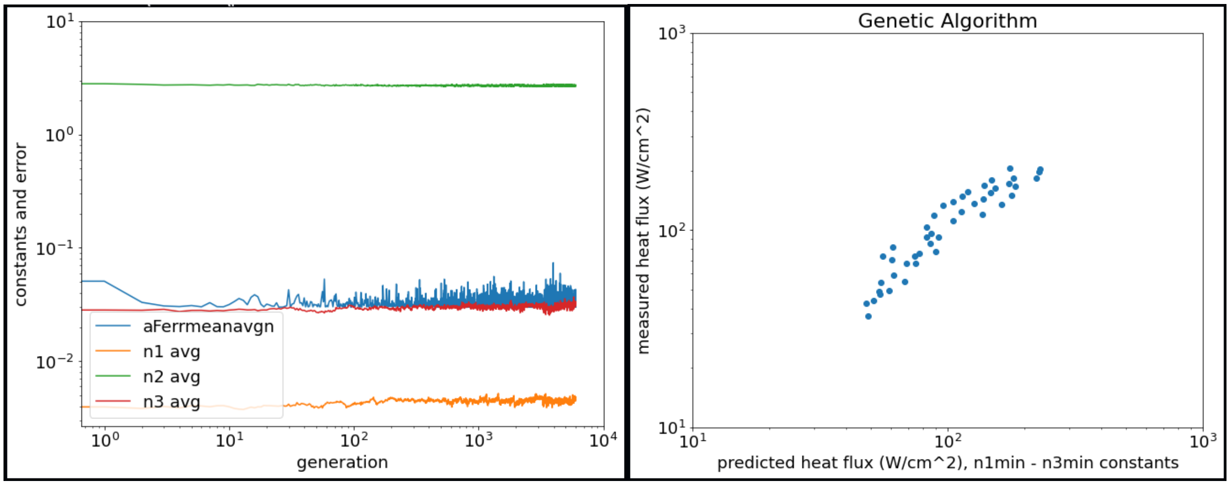

Figure 5: Trial 2 Constants and Error and Measured Heat Flux vs Predicted Heat Flux

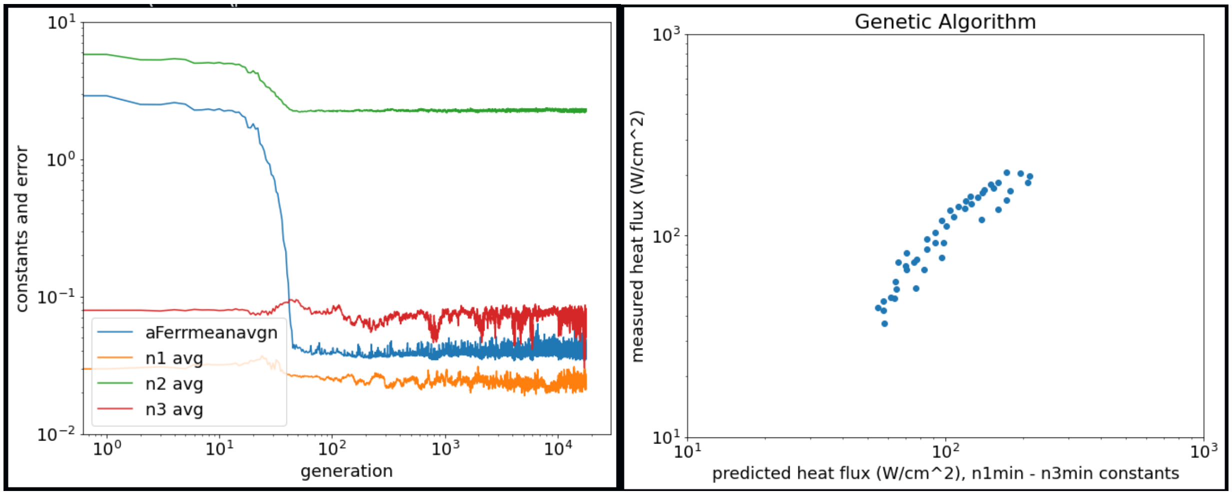

Figure 6: Trial 3 Constants and Error and Measured Heat Flux vs Predicted Heat Flux

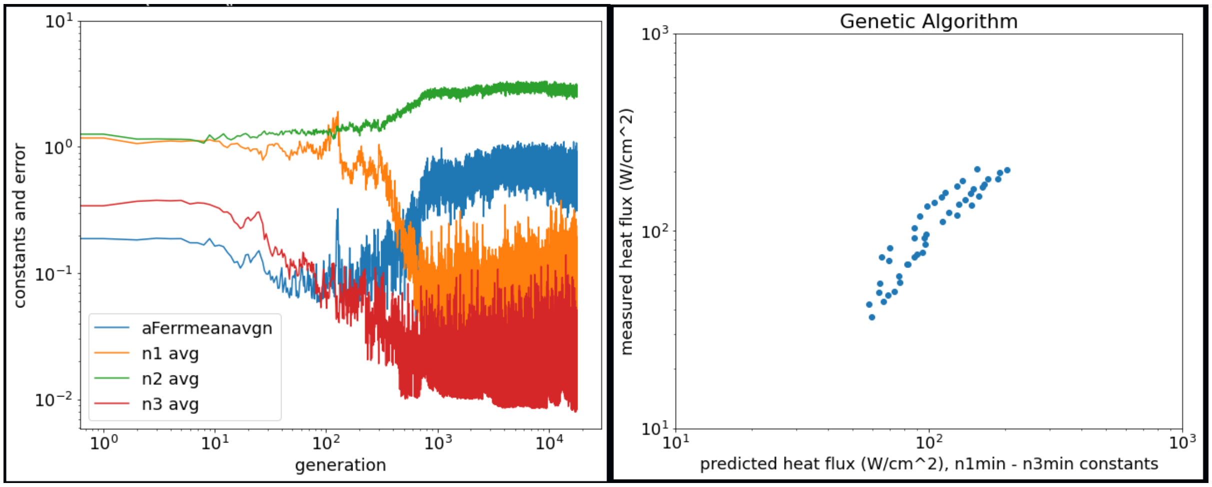

Figure 7: Trial 4 Constants and Error and Measured Heat Flux vs Predicted Heat Flux

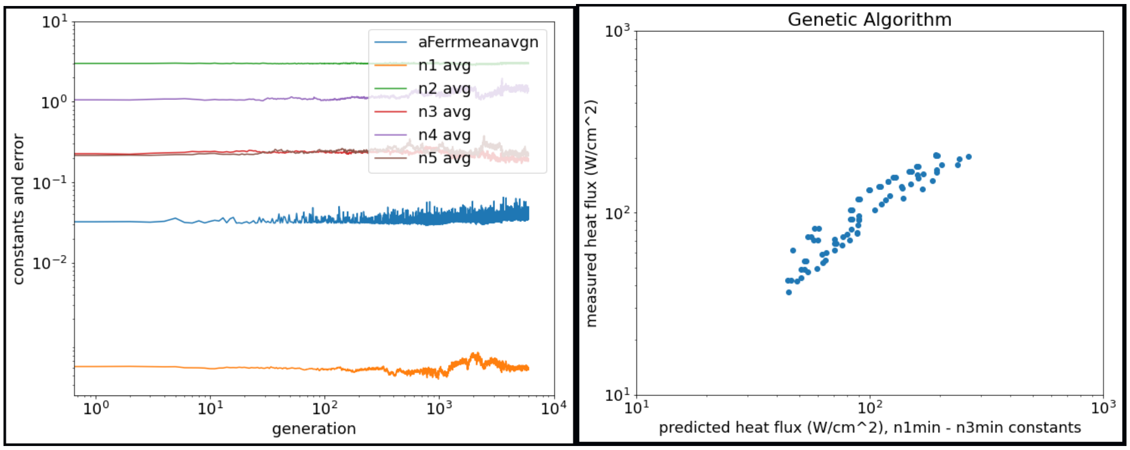

Figure 8: Task 3 Constants and Error and Measured Heat Flux vs Predicted Heat Flux

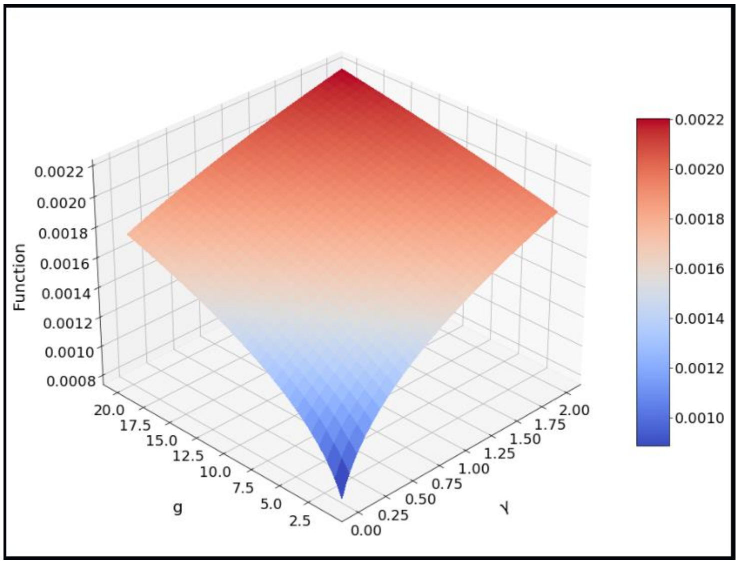

Figure 9: Surface Plot of $q''/(T_w - T_{sat})^{n2}$ *vs.* g *and* γ

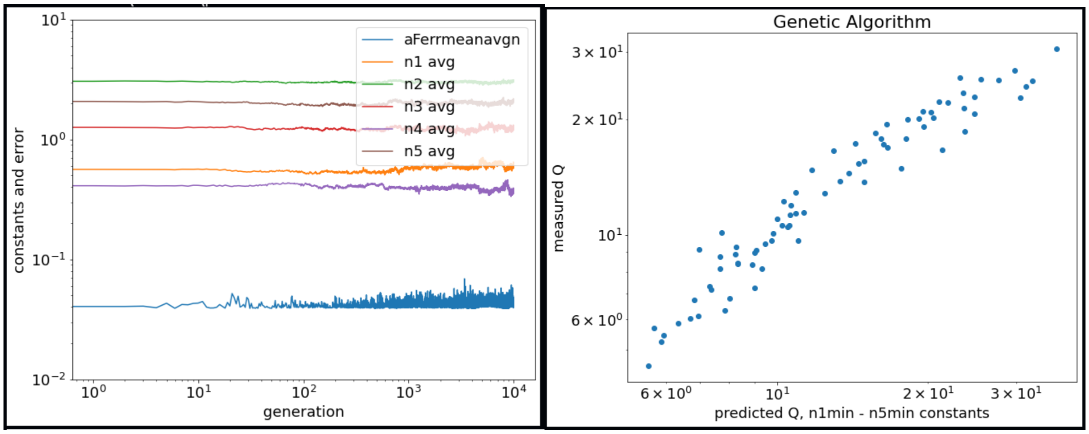

Figure 10: Task 5.a Constants and Error and Measured Q vs Predicted Q

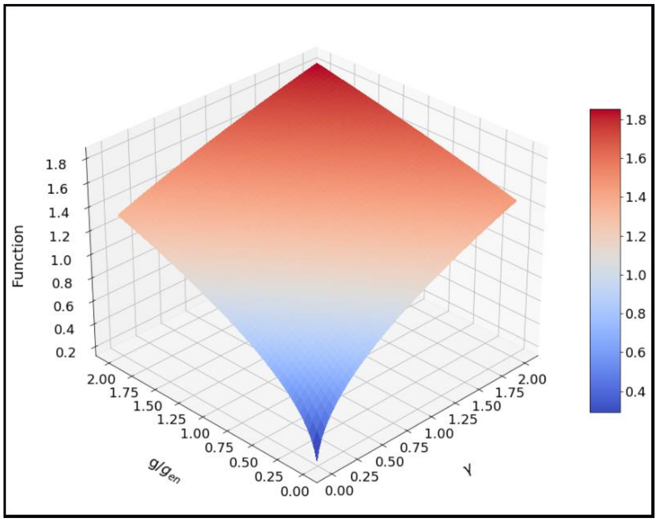

Figure 11: Surface Plot of $Q_s\, Pr_l^{n5}/Ja_s^{n2}$ *vs.* g/g_{en} *and* γ

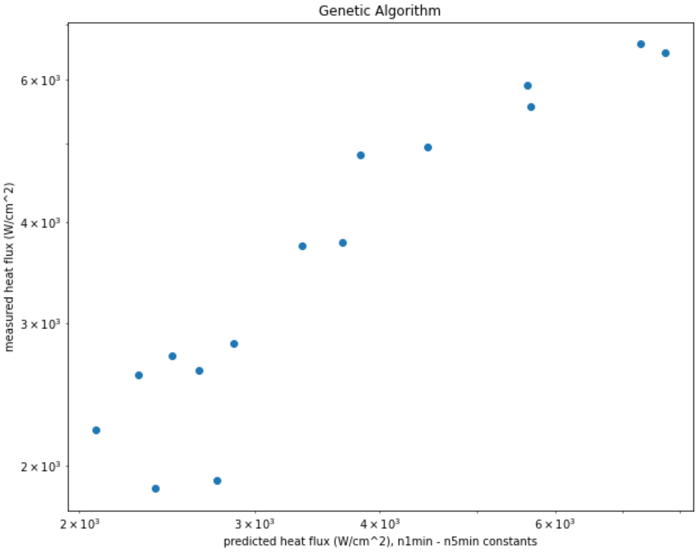

Figure 12: Task 5.c plot between Measured q" vs Predicted q"

2. Outputs

```
ydata = [[44.1, 32.5, 0.098, 1.79, 5.5], [47.4, 33.2, 0.098, 1.79, 5.5], [49.4, 34.2, 0.098, 1.79, 5.5], [59.2,
34.8, 0.098, 1.79, 5.5], [67.8, 36.3, 0.098, 1.79, 5.5], [73.6, 37.3, 0.098, 1.79, 5.5], [76.3, 37.8, 0.098, 1.79,
5.5], [85.3, 39.2, 0.098, 1.79, 5.5], [96.5, 39.3, 0.098, 1.79, 5.5], [111.0, 42.3, 0.098, 1.79, 5.5], [124.0, 43.5,
0.098, 1.79, 5.5], [136.2, 45.4, 0.098, 1.79, 5.5], [143.5, 46.7, 0.098, 1.79, 5.5], [154.6, 47.9, 0.098, 1.79,
5.5], [163.1, 48.6, 0.098, 1.79, 5.5], [172.8, 50.9, 0.098, 1.79, 5.5], [184.2, 51.7, 0.098, 1.79, 5.5], [203.7,
56.4, 0.098, 1.79, 5.5], [36.7, 30.2, 9.8, 1.79, 5.5], [55.1, 34.1, 9.8, 1.79, 5.5], [67.5, 35.3, 9.8, 1.79, 5.5],
[78.0, 37.8, 9.8, 1.79, 5.5], [92.0, 38.1, 9.8, 1.79, 5.5], [120.0, 44.1, 9.8, 1.79, 5.5], [134.3, 46.9, 9.8, 1.79,
5.5], [150.3, 48.5, 9.8, 1.79, 5.5], [167.0, 49.2, 9.8, 1.79, 5.5], [184.0, 52.7, 9.8, 1.79, 5.5], [196.5, 53.1, 9.8,
1.79, 5.5], [42.4, 29.7, 19.6, 1.79, 5.5], [48.7, 31.0, 19.6, 1.79, 5.5], [54.5, 31.2, 19.6, 1.79, 5.5], [70.8, 32.4,
19.6, 1.79, 5.5], [73.7, 31.4, 19.6, 1.79, 5.5], [81.8, 32.5, 19.6, 1.79, 5.5], [91.9, 36.3, 19.6, 1.79, 5.5], [103.9,
36.3, 19.6, 1.79, 5.5], [119.1, 37.2, 19.6, 1.79, 5.5], [133.7, 38.4, 19.6, 1.79, 5.5], [139.9, 39.7, 19.6, 1.79,
5.5], [148.3, 40.9, 19.6, 1.79, 5.5], [157.0, 41.6, 19.6, 1.79, 5.5], [169.1, 43.9, 19.6, 1.79, 5.5], [179.2, 45.0,
19.6, 1.79, 5.5], [205.0, 47.9, 19.6, 1.79, 5.5]]
```

Output 1 : CodeP1.1 ydata

```
ydata =
```

```
[5.463213627095972, 5.618279569892473, 0.01, 1.79, 4.83]
[5.872025531164378, 5.739288668320927, 0.01, 1.79, 4.83]
[6.119790321508867, 5.912158808933002, 0.01, 1.79, 4.83]
[7.33383779419686, 6.015880893300247, 0.01, 1.79, 4.83]
[8.39922639267816, 6.275186104218361, 0.01, 1.79, 4.83]
[9.117744284677178, 6.448056244830439, 0.01, 1.79, 4.83]
[9.452226751642238, 6.534491315136476, 0.01, 1.79, 4.83]
[10.567168308192436, 6.776509511993383, 0.01, 1.79, 4.83]
[11.95465113412157, 6.79379652605459, 0.01, 1.79, 4.83]
[13.750945864119112, 7.312406947890818, 0.01, 1.79, 4.83]
[15.361417001358289, 7.5198511166253095, 0.01, 1.79, 4.83]
[16.872782222459666, 7.848304383788254, 0.01, 1.79, 4.83]
[17.77712370721705, 8.073035566583952, 0.01, 1.79, 4.83]
[19.152218293628962, 8.280479735318444, 0.01, 1.79, 4.83]
[20.20521865259304, 8.401488833746898, 0.01, 1.79, 4.83]
[21.406877885763812, 8.799090157154673, 0.01, 1.79, 4.83]
[22.81913719072739, 8.937386269644335, 0.01, 1.79, 4.83]
[25.234843896586153, 9.749875930521092, 0.01, 1.79, 4.83]
[4.546483902821365, 5.220678246484697, 1.0, 1.79, 4.83]
[6.825919973990659, 5.894871794871794, 1.0, 1.79, 4.83]
[8.362061674126487, 6.1023159636062845, 1.0, 1.79, 4.83]
[9.662826823435053, 6.534491315136476, 1.0, 1.79, 4.83]
[11.397180355846471, 6.586352357320098, 1.0, 1.79, 4.83]
[14.865887420669312, 7.623573200992556, 1.0, 1.79, 4.83]
[16.637405671632408, 8.107609594706368, 1.0, 1.79, 4.83]
[18.619523994388313, 8.38420181968569, 1.0, 1.79, 4.83]
[20.68835999376479, 8.505210918114143, 1.0, 1.79, 4.83]
[22.794360711692942, 9.11025641025641, 1.0, 1.79, 4.83]
[24.342890651345996, 9.179404466501241, 1.0, 1.79, 4.83]
[6.331242458827516, 4.888888888888888, 2.0, 1.79, 3.91]
[7.271969522285379, 5.1158730158730155, 2.0, 1.79, 3.91]
[8.138035707691028, 5.168253968253968, 2.0, 1.79, 3.91]
[9.27288105408464, 4.976190476190475, 2.0, 1.79, 3.91]
[10.571980332193116, 5.325396825396825, 2.0, 1.79, 3.91]
[11.00501342489594, 5.29047619047619, 2.0, 1.79, 3.91]
[12.214519649341762, 5.3428571428571425, 2.0, 1.79, 3.91]
[13.722669385996433, 6.023809523809523, 2.0, 1.79, 3.91]
[15.514530459249505, 6.023809523809523, 2.0, 1.79, 3.91]
[17.784221152036725, 6.18095238095238, 2.0, 1.79, 3.91]
[19.964318791161293, 6.425396825396825, 2.0, 1.79, 3.91]
[20.890113679008714, 6.652380952380951, 2.0, 1.79, 3.91]
[22.14441643028586, 6.826984126984126, 2.0, 1.79, 3.91]
[23.443515708394337, 6.984126984126984, 2.0, 1.79, 3.91]
[25.25030895725785, 7.368253968253969, 2.0, 1.79, 3.91]
[26.75845869391252, 7.542857142857143, 2.0, 1.79, 3.91]
[30.610960001406625, 8.03174603174603, 2.0, 1.79, 3.91]
[5.2526135553031565, 5.134243176178659, 2.0, 1.79, 4.83]
[6.033072644888296, 5.358974358974359, 2.0, 1.79, 4.83]
[6.751590536887313, 5.393548387096774, 2.0, 1.79, 4.83]
[8.770873578194895, 5.600992555831265, 2.0, 1.79, 4.83]
[9.130132524194401, 5.428122415219189, 2.0, 1.79, 4.83]
[10.13357992508958, 5.618279569892473, 2.0, 1.79, 4.83]
[11.384792116329248, 6.275186104218361, 2.0, 1.79, 4.83]
[12.87138085839618, 6.275186104218361, 2.0, 1.79, 4.83]
```

```
[14.75439326501429, 6.430769230769232, 2.0, 1.79, 4.83]
[16.563076234529056, 6.638213399503721, 2.0, 1.79, 4.83]
[17.331147084596974, 6.862944582299421, 2.0, 1.79, 4.83]
[18.371759204043823, 7.070388751033913, 2.0, 1.79, 4.83]
[19.44953604204235, 7.1913978494623665, 2.0, 1.79, 4.83]
[20.948513023626504, 7.58899917287014, 2.0, 1.79, 4.83]
[22.19972521486617, 7.779156327543424, 2.0, 1.79, 4.83]
[25.395891010310073, 8.280479735318444, 2.0, 1.79, 4.83]
[10.45556972006653, 7.209891936824605, 1.0, 0.0, 4.54]
[9.640850001619787, 7.0361596009975065, 1.0, 0.0, 4.54]
[8.961916902914169, 6.862427265170407, 1.0, 0.0, 4.54]
[8.418770423949674, 6.688694929343307, 1.0, 0.0, 4.54]
[5.703038029127198, 5.906899418121363, 1.0, 0.0, 4.54]
[8.147197184467426, 6.514962593516209, 1.0, 0.0, 4.54]
[7.19669084627956, 6.42809642560266, 1.0, 0.0, 4.54]
[8.882367733849915, 6.2924731182795695, 0.01, 1.71, 4.83]
[10.096415206537907, 6.655500413564928, 0.01, 1.71, 4.83]
[11.236133242122555, 6.828370554177004, 0.01, 1.71, 4.83]
[12.79705142129283, 7.1913978494623665, 0.01, 1.71, 4.83]
[14.494240235152578, 7.450703060380478, 0.01, 1.71, 4.83]
[17.170099970873054, 7.848304383788254, 0.01, 1.71, 4.83]
[20.031783299351897, 8.280479735318444, 0.01, 1.71, 4.83]
[25.705596998240683, 8.799090157154673, 0.01, 1.71, 4.83]
```

Output 2:Task 4: ydata printed in table format

3. Table

Trial	**Gene n1i**	**Gene n2i**	**Gene n3i**	**NGEN**	**MFRAC**	**MR**	**Min error**
Base	0.00027	4.0	0.063	6000	0.5	0.09	0.031
1	0.029	4.6	0.03	6000	0.8	0.25	0.030
2	0.0035	2.7721	0.02836	6000	0.4	0.06	0.0293
3	0.029	5.6	0.0788	18000	0.8	0.07	0.0307
4	1.1	1.2	0.35	18000	0.9	0.4	0.038

Table 1: Task Two Trials and Respective Inputs

	P = 5.5 kPa	P = 7.0 kPa	P = 9.5 kPa
T_{sat} ($°C$)	34.9	38.0	45.0
c_{pl} (kJ/kg°C)	4.18	4.18	4.18
h_{lv} (kJ/kg°C) 2	2418	2406	2394

μ_l (Ns/m^2)	7.19x10^{-4}	6.53x10^{-4}	5.96x10^{-4}
Pr_l	4.83	4.54	3.91
ρ_l (kg/m^3)	994	993	990
ρ_v (kg/m^3)	0.0397	0.0476	0.182
σ (N/m)	0.0706	0.0692	0.0688

Table 2: Low Pressure Water Saturation Properties