**Lab 1: Interpolation and Scaling**

Disclosure: Although the base of this lab is rooted in a potentially real situation, the data is fabricated and in no way is intended to represent the true relationship between temperature of water in a curing bladder and deflection of a tire.

Objectives

* Understand the effect of interpolation method on results
* Understand the advantages/disadvantages in gathering more data
* Practice using interpolation and scaling in MATLAB

View the video at the following link on how tires are made:

<https://www.youtube.com/watch?v=K474RYse9P8>

From this video it is obvious that the process to make a tire is very complex and there are many factors that affect the final quality of a tire. There are also multiple ways to measure the quality of a tire (ex. life, traction, deflection). To simplify this lab, we are going to focus on 1 parameter of the manufacturing process (the temperature of the hot pressurized water that is injected into the curing bladder) and on one measure of quality (the deflection in the tire with certain inflation pressure and under a given load).

In the curing mold, materials change from a plastic to an elastic state. This reaction is effected by multiple factors including the temperature of the hot pressurized water injected into the mold. Tire companies are very secretive about their processes and even the engineers do not know the real temperature at which the process is running. The temperatures have been coded and are referred to as AA Tire degrees.

You have landed your first co-op position, working for AA Tire Company, and your boss has asked you to create a profile of the relationship between the temperature of the hot pressurized water in the curing bladder and the deflection of the tire. By profile, she means what is the deflection of the tire for different temperatures of water.

For the same tire there are multiple factors that will impact deflection. The inflation air pressure in the tire (measured in PSI) and the load on the tire (measured in tons) are the biggest concerns since other factors, such as temperature, will be controlled for in the lab.

Your boss asks you to think about how you would do this project and to come back with questions. Can you think of what questions you would ask your boss?

Here are some questions I would ask based on the problem solving step - **defining criteria**:

* What will the profile be used for?
* How accurate does the data need to be?
* What range does the profile need to cover?
* What are the set points in the experiment of the factors other than water temperature? (ex. pressure of water, which type of tire (since AA Tire makes more than one type of tire))
* What test conditions should you use? What load and inflation pressure?

Here are some questions I would ask based on the problem solving step - **defining constraints**:

* What is my budget? What account number do I charge material to?
* When do you need the profile?
* Do I have any additional employees that can help?

This is what you find out from asking these questions:

* The data will be used to try to determine the amount of deflection in the tire for a given temperature. If you did not run the tire with that exact temperature, the answer will be interpolated from the data you did gather.
* Since the deflection of the tire is a final quality check, having the process centered at the desired deflection is critical so the data needs to accurate, but you must operate within your budget.
* You are given $800 in supplies and 1 week, but your boss asks you to spend as little money and time as possible.
* You are given the account number to charge the supplies to.
* You have to do all the work yourself, you are the co-op after all.
* You are told what set points to use for the other variables (will not be needed for this lab), but your boss asks you to figure out the temperature range to run the experiment under.
* For testing you go with the high end of the rating for load, since this would be the peak deflection. You decide to go with the low range of the inflation pressure.

**Part 1.**

**Since your boss did not tell you what range to run the profile for, how would you decide?**

|  |
| --- |
| Research values that make sense (a sensible low and high value) and try to minimize the range operating in while ensuring an acceptable sample size. |

**What is the disadvantage of using too large a range?**

|  |
| --- |
| Some values at extremes can lead to misleading results that could throw off the data set. |

**What is the disadvantage of using too small a range?**

|  |
| --- |
| If relevant data is left out, the data set may not be well representative of the desired results. |

First thing you do is calculate the cost and time for each experimental run. You have access to nonproduction equipment so you do not have to shut down the production line to run your experiment. Assume the following:

* Pre-work Time: Design time, securing materials, checking with the lab to make sure the equipment is available, etc. - 10 hrs.
* Labor-hours/run: labor to prepare, run test, measure results - 3 hrs. per run
* Cost of supplies per run: $50

Your boss has cleared you of all your other responsibilities and has given you one week (40 hours) to complete the pre-work and experimental runs.

**Based on the cost of supplies per run and the total budget allotted, how many runs can you make (show your work)?**

|  |
| --- |
| 800/50 = 16 runs |

**Based on the time requirements for pre-work and the time per run, what is the maximum number of runs you can complete in the given time frame (show your work)?**

|  |
| --- |
| (40 – 10)/3 = 10 runs |

**To meet both constraints, how many runs will you be able to make? Why?**

|  |
| --- |
| 10 runs because that is all time will allow even though the budget allows for more. |

**Do you think this is enough?**

|  |
| --- |
| While having more quality data is always good, 10 runs should be sufficient especially considering the accessibility to interpolation. |

After talking to people familiar with the process and checking standard operating procedures, you decide to use a **range from -10 to 10.0 AA Tire degrees.**

For this lab: Since you cannot actually run the experiment and determine the deflection of the tire, data has been supplied to you. Data file tire\_lab.xls contains the results of many more values of temperature than you can afford to run, so you must only pull out the number of data points that you can afford to run. If you decide you can afford to run *n* runs then decide which *n* data points within the data set to use. The data runs from x = -15 to x = 15 AA Tire degrees.

**Part 2.**

Import the data file **tire\_lab.xls** into MATLAB and take a look at your data. Notice the data points are arranged from lowest temperature to highest.

Start a new MATLAB script file. Determine how to get your data points from the file then add the commands to your script file to extract the desired data points. Since you decided to run between -10 and 10 AA Tire degrees, no points should have the temperature under -10 or over 10. There are many ways to do this.

*Hint: Once you know which row has the -10 and 10 AA Temp degree you can use floor(linspace(r1,r2,N)) where r1 = the row with -10 and r2 = the row with 10. This will give you the row locations for N points which span the range are not quite equally spaced but close. Use these locations to get your AA Tire temperatures and the deflection values.*

Throughout this lab, you will need to display results in tabular form. Here is an **example** of one way to print a table using fprintf. \t is used like a ‘Tab’ button on your computer*. THIS IS JUST AN EXAMPLE – TRY IT IN THE COMMAND WINDOW BUT YOU WILL NEED TO SIGNIFICANTLY MODIFY IT IN YOUR SCRIPT FILE.*

x = [6.65 7.5]; y = [-1.56 7.3];

fprintf('\ncolumn 1 title\tcolumn 2 title\tcolumn 3 title\n');

fprintf('%14s\t%14.4f\t%14.4f\n','row 2 title',x(1),y(1));

fprintf('%14s\t%14.4f\t%14.4f\n','row 3 title',x(2),y(2));

An alternative to a series of fprintf statements is the ***table*** command. At the MATLAB command prompt, type: doc table then expand the examples section to see how the ***table*** command works.

In your script, add commands to display the data points (AA Tire degrees and deflections) to your command window as a table**.  *In the space below the header for Table 1, copy and paste your output for the data you selected****.*

**Table 1. Data Points Taken from tire\_lab.xls**

AA\_Tire\_Degrees Deflection

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_

-10 139

-8 129

-6 119

-4 79

-2 39

1 13

3 14

5 72

7 141

10 261

Add commands to your script file that will run linear, nearest point and spline interpolation to estimate the tire deflection for the AA Tire temperatures listed in Table 2 using only the data points in Table 1. **Hint: you are going to run this again for a different number of data points so program accordingly.** Within the script, calculate the percent error and the absolute value of the error using the actual values for deflection provided in the tire\_lab.xls file.

Percent Error = abs(estimate-actual)/actual\*100%

Within the script, use **fprintf** statements or the **table** command to output a table containing for each temperature, the 3 estimates, the actual value, the percent error and the absolute value of the error for the estimated values.

Fill out Table 2 or just paste your table from MATLAB.

**Table 2. Interpolation Estimates, Actual Deflection and Percent Error**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| AA Tire Temperature | Nearest estimate  (mm) | Linear estimate  (mm) | Spline estimate  (mm) | Actual  (mm) | Percent error Nearest | Percent error Linear | Percent error Spline |
| -9.7 | 139 | 137.5 | 135.68 | 136 | 2.2 | 1.1 | 0.23 |
| -6.7 | 119 | 122.5 | 125.38 | 129 | 7.7 | 5.0 | 2.7 |
| .2 | 13 | 19.93 | 17.87 | 13 | 0 | 53.3 | 37.5 |
| 4.3 | 72 | 51.7 | 47.63 | 60 | 20 | 13.8 | 20.6 |
| 9.5 | 261 | 241 | 238.01 | 241 | 8.2 | 0 | 1.2 |

|  |  |  |  |
| --- | --- | --- | --- |
| AA Tire Temperature | Absolute value of error Nearest (mm) | Absolute value of error Linear (mm) | Absolute value of error Spline (mm) |
| -9.7 | 3 | 1.5 | 0.31 |
| -6.7 | 10 | 6.5 | 3.6 |
| .2 | 0 | 6.93 | 4.8 |
| 4.3 | 12 | 8.3 | 12.3 |
| 9.5 | 20 | 0 | 2.9 |

Within your script, calculate the mean of the absolute value of the errors and mean square error for each type of interpolation. Mean square error is equal to the mean of the errors squared. Enter the results in Table 3 or paste table from MATLAB.

**Table 3. Mean of Absolute Value of Errors and Mean Square Error**

**For Each Interpolation Method**

|  |  |  |
| --- | --- | --- |
|  | Mean of the absolute value of the errors | Mean Square Error |
| Nearest | 9 | 130.6 |
| Linear | 4.64 | 32.29 |
| Spline | 4.83 | 39.76 |

On the same graph, plot the points for each type of interpolation (in Table 2 above), the points that were used to interpolate (in Table 1 above) and all the data for degrees and deflection in the data set. Make sure you use different symbols for each type of data and include a legend and paste your graph below.

**PASTE GRAPH HERE:**



**What do you notice about the results? Which method had the lowest mean square error and mean of absolute value of errors?**

|  |
| --- |
| The linear estimate has the least amount of error according to the results. |

**Part 3.**

Repeat Part 2, but this time use twice as many points from tire\_lab.xls.

**PASTE ALL TABLES AND GRAPHS HERE:**

AA\_Tire\_Degrees Deflection

-10 139

-9 129

-8 129

-7 129

-6 119

-5 99

-4 79

-3 59

-2 39

-1 23

0 13

1 13

2 14

3 14

4 48

5 72

6 101

7 141

8 181

10 261

interp\_points nearest\_estimate linear\_estimate spline\_estimate actual per\_error\_nearest per\_error\_linear per\_error\_spline abs\_error\_nearest abs\_error\_linear abs\_error\_spline

-9.7 139 136 134.38 136 2.2059 0 1.1887 3 0 1.6167

-6.7 129 126 127.29 129 0 2.3256 1.3229 0 3 1.7065

0.2 13 13 12.278 13 0 0 5.5553 0 0 0.72219

4.3 48 55.2 56.664 60 20 8 5.5607 12 4.8 3.3364

9.5 261 241 240.22 241 8.2988 0 0.32529 20 0 0.78396

mean\_error\_nearest mean\_error\_linear mean\_error\_spline

Abs Error Mean 7 1.56 1.6332

Error Squared Mean 110.6 6.408 3.5588

****

**What did you notice about the percent error for this experiment as compared to when you used half as many points?**

|  |
| --- |
| The linear still has a slightly better abs error but the spline now has a much better squared error. |

**PASTE MATLAB SCRIPT HERE:**

%Creator: Jonathan Kenney

%Lab 1: Interpolation

clear; clc; close all;

data\_raw = xlsread('tire\_lab.xlsx');

r1 = -10;

r2 = 10;

N = 20;

tire\_temps = floor(linspace(r1,r2,N));

data = zeros(N, 3);

for k = 1:N

p = find(data\_raw(:,1) == tire\_temps(k));

data(k,:) = data\_raw(p,:);

end

AA\_Tire\_Degrees = data(:,1);

Deflection = data(:,2);

T1 = table(AA\_Tire\_Degrees, Deflection);

disp(T1);

interp\_points = [-9.7, -6.7, .2, 4.3, 9.5];

nearest\_estimate = interp1(data(:,1),data(:,2),interp\_points, 'nearest');

linear\_estimate = interp1(data(:,1),data(:,2), interp\_points, 'linear');

spline\_estimate = interp1(data(:,1),data(:,2), interp\_points, 'spline');

actual = zeros(1,length(interp\_points));

for k = 1:length(interp\_points)

p = find(data\_raw(:,1) == interp\_points(k));

actual(k) = data\_raw(p,2);

end

per\_error\_nearest = zeros(1,length(interp\_points));

per\_error\_linear = zeros(1,length(interp\_points));

per\_error\_spline = zeros(1,length(interp\_points));

abs\_error\_nearest = zeros(1,length(interp\_points));

abs\_error\_linear = zeros(1,length(interp\_points));

abs\_error\_spline = zeros(1,length(interp\_points));

for k = 1:length(interp\_points)

per\_error\_nearest(k) = abs(nearest\_estimate(k) - actual(k))/actual(k)\*100;

abs\_error\_nearest(k) = abs(nearest\_estimate(k) - actual(k));

end

for k = 1:length(interp\_points)

per\_error\_linear(k) = abs(linear\_estimate(k) - actual(k))/actual(k)\*100;

abs\_error\_linear(k) = abs(linear\_estimate(k) - actual(k));

end

for k = 1:length(interp\_points)

per\_error\_spline(k) = abs(spline\_estimate(k) - actual(k))/actual(k)\*100;

abs\_error\_spline(k) = abs(spline\_estimate(k) - actual(k));

end

interp\_points = interp\_points';

nearest\_estimate = nearest\_estimate';

linear\_estimate = linear\_estimate';

spline\_estimate = spline\_estimate';

actual = actual';

per\_error\_nearest = per\_error\_nearest';

per\_error\_linear = per\_error\_linear';

per\_error\_spline = per\_error\_spline';

abs\_error\_nearest = abs\_error\_nearest';

abs\_error\_linear = abs\_error\_linear';

abs\_error\_spline = abs\_error\_spline';

T2 = table(interp\_points, nearest\_estimate, linear\_estimate, spline\_estimate, actual,...

per\_error\_nearest, per\_error\_linear, per\_error\_spline,...

abs\_error\_nearest, abs\_error\_linear, abs\_error\_spline);

disp(T2);

interp\_points = interp\_points';

nearest\_estimate = nearest\_estimate';

linear\_estimate = linear\_estimate';

spline\_estimate = spline\_estimate';

actual = actual';

abs\_error\_nearest = abs\_error\_nearest';

abs\_error\_linear = abs\_error\_linear';

abs\_error\_spline = abs\_error\_spline';

mean\_error\_nearest = [sum(abs\_error\_nearest)/length(abs\_error\_nearest) sum(abs\_error\_nearest.^2)/length(abs\_error\_nearest)]';

mean\_error\_linear = [sum(abs\_error\_linear)/length(abs\_error\_linear) sum(abs\_error\_linear.^2)/length(abs\_error\_linear)]';

mean\_error\_spline = [sum(abs\_error\_spline)/length(abs\_error\_spline) sum(abs\_error\_spline.^2)/length(abs\_error\_spline)]';

T3RowNames = {'Abs Error Mean'; 'Error Squared Mean'};

T3 = table(mean\_error\_nearest, mean\_error\_linear, mean\_error\_spline, 'RowNames', T3RowNames);

disp(T3);

mean\_error\_nearest = mean\_error\_nearest';

mean\_error\_linear = mean\_error\_linear';

mean\_error\_spline = mean\_error\_spline';

plot(AA\_Tire\_Degrees,Deflection,'rx',interp\_points,nearest\_estimate,'b\*',...

interp\_points,linear\_estimate,'k\*',interp\_points,spline\_estimate,'y\*',...

data\_raw(:,1),data\_raw(:,2));

legend('Selected Deflections','Nearest Estimate','Linear Estimate','Spline Estimate','Raw Data');

xlabel('Tire Degrees');

ylabel('Deflection');

**Part 4.**

Years later after you have graduated from UC and you have been working with AA Tire Company you have been promoted to top engineer and you now have access to the coding between AA Tire degrees and oC. You are now responsible to make sure the thermocouples are calibrated correctly and that the AA Tire degrees corresponds to the correct oC. The secret formula is that -10 AA Tire degrees = 130 oC, 10 AA Tire degrees = 210 oC, and the relationship is linear.

**Write the equation for the formula to calculate oC from AA Tire degrees.**

|  |
| --- |
| y = 4x + 170 where y is degrees in Celsius and x is AA Tire degrees. |

For each of the twenty AA Tire Degrees you selected in Part 3, calculate the oC from the formula you just derived. Pull the actual temperatures (oC) from the **tire\_lab.xls** file. On the same graph, plot the actual and calculated temperatures (oC) vs. the AA Tire Degrees (x-axis).

**PASTE GRAPH HERE:**

****

**Do the thermocouples need to be calibrated? If so how big of an adjustment needs to be made?**

|  |
| --- |
| Yes they need to be adjusted, and they are reporting four degrees too high. |

Clark, S. (1981). *Mechanics of pneumatic tires*. Washington DC: U.S. Department of Transportation, National Highway Traffic Safety Administration

<https://books.google.com/books?id=zcZTAAAAMAAJ&pg=PA225&lpg=PA225&dq=plastic+to+elastic+state+in+a+tire&source=bl&ots=7LMtx_BjKR&sig=cST8_nbRtkIs5eNiqsXhf5JFx4c&hl=en&sa=X&ved=0ahUKEwjFid3O2LjKAhUJZCYKHUHoCysQ6AEIKjAD#v=onepage&q=plastic%20to%20elastic%20state%20in%20a%20tire&f=false>