Math 104A Final Project: Approximating the Specific Heat Required to Raise Water Temperature by 1°C

Kameron Quackenbush, 8868416
12/8/19
Group Members: Shawden Soltani, Niko Kalpakian, Amit Yativ, Emma Furr, Kelcie Opp

MOTIVATION

For the Final Project, our group decided to analyze the specific heat of water. We wanted to approximate how much specific heat would be required to raise the temperature of water by 1° Celsius at some given temperature. Using an existing data set that has the specific heat of water at temperatures ranging from $0 - 100$°C.

Background

Heat capacity or thermal capacity (usually denoted by a capital C) is a physical property of matter. Heat capacity is the measurable physical quantity that characterizes the amount of heat required to change a substance's temperature by a given amount. It is measured in Joules per Kelvin (J/K). The heat capacity of most systems is not constant. In our project, we are dealing with water in a constant-pressure environment. The specific heat is an intensive property that describes how much heat must be added to a particular substance to raise its temperature. Unlike the total heat capacity, the specific heat capacity is independent of mass or volume. It describes how much heat must be added to a unit of mass of a given substance to raise its temperature by one degree Celsius. The units of specific heat capacity are J/(kg °C) or equivalently J/(kg K). The heat capacity and the specific heat are related by $C = cm$ or $c = C/m$, where C denotes heat capacity, c denotes specific heat, and m denotes mass. The mass m, specific heat c, change in temperature $\triangle T$, and heat added (or subtracted) Q are related by the equation: $Q = mc\triangle T$.

Problem to Solve

Values of specific heat cannot be calculated easily, and generally must be looked up in reference tables. In general, the specific heat also depends on temperature. Due to these difficulties, our goal was to create interpolating polynomials that allow us to approximate specific heat values needed to raise the temperature of water by 1°C at different given temperatures. Our main question that we planned to answer is which approximation method is the most effective in determining the specific heat needed to raise the temperature of water by 1°C at a certain temperature?

METHODS AND PROCEDURES

Overview

Table 1

Temperature (°C)	Specific Heat (J/kg-C)
20.00	4182.00
40.00	4179.00
50.00	4182.00
80.00	4198.00
90.00	4208.00

Table 1

Table 2

Temperature (°C)	Specific Heat (J/kg-C)
10.00	4192.00
30.00	4178.00
50.00	4182.00
70.00	4191.00
90.00	4208.00

Table 2

In this project, we will use Neville's method, Lagrange Interpolation, and the Natural Cubic Spline method. All three of these methods are utilized to find interpolating polynomials to approximate functions at specific nodes. We are given measured data, reference to *Table 1* and *Table 2* above, and use it to approximate data that we do not have. We are approximating an Isobaric (i.e. constant pressure (C*p*) specific heat. Our unit for specific heat is J/(kg °C). For this project we are going to approximate the specific heat for $x = 25°C$ and for $x = 85°C$

Method One: Natural Cubic Spline

Natural cubic spline method interpolates a set of data points with piece-wise cubic polynomials. Theorem 3.11 from the book states: If f is defined at $a = x_0 < x_1 < ... < x_n$; then f has a unique natural spline interpolant S on the nodes $x_0, x_1, ..., x_n$; that is, a spline interpolant that satisfies the natural boundary conditions $S''(a) = 0$ and $S''(b) = 0$.

We used this method first, following theorem 3.11 and through implementation of algorithm 3.4 in the book, predicting that it will be the most accurate interpolation method. The other methods concerned the approximation of arbitrary methods on closed intervals using a single polynomial. The Natural Cubic Spline Method allows us to divide the approximation interval into a collection of sub-intervals and construct a (generally) different approximating polynomial on each sub-interval. Therefore, our Natural Cubic spline has less tendency to oscillate between data points compared to a polynomial.

Method Two: Fourth Degree Lagrange Interpolation

The Lagrange interpolation formula is a way to find a polynomial which takes on certain values at arbitrary points. Theorem 3.2 in the book states:

If $x_0, x_1, ..., x_n$ are $n + 1$ distinct numbers, and f is a function whose values are given at these numbers, then a unique polynomial $P(x)$ of degree at most n exists with

$$f(x_k) = P(x_k), \text{ for each } k = 0, 1, ..., n.$$

This polynomial is given by

$$P(x) = f(x_0)L_{n,0}(x) + ... + f(x_n)L_{n,n}(x) = \sum_{k=0}^{n} f(x_{n,k})L_{n,k}(x).$$

Where, for each $k = 0, 1, ..., n$,

$$L_{n,k}(x) = \frac{(x - x_0)(x - x_1)...(x - x_{k-1})(x - x_{k+1})...(x - x_n)}{(x_k - x_0)(x_k - x_1)...(x_k - x_{k-1})(x_k - x_{k+1})...(x_k - x_n)}$$

$$= \prod_{i=0, i\neq k}^{n} \frac{(x - x_i)}{(x_k - x_i)}.$$

In our case, we had two data sets with five data points each, allowing us to come up with two fourth-degree Lagrange interpolating polynomials. We used this method second, predicting a less accurate interpolation than the Cubic Spline.

Method Three: Neville's Method

The final method we used was Neville's method, which can be applied when we want to interpolate *f(x)* at a given point *x=p* with increasingly higher order Lagrange interpolating polynomials. Thus, we predicted getting results similar to the results using Lagrange. In Neville's method, we use the Lagrange result from Theorem 3.5 in the book to recursively generate interpolating polynomial approximations. Theorem 3.5 can be seen below:

Let f be defined at $x_0, x_1, ..., x_k$ and let x_j and x_i be two distinct numbers in this set. Then,

$$P(x) = \frac{(x - x_j)P_{0,1,...j-1,j+1,...,k}(x) - (x - x_i)P_{0,1,,,j-1,j+1,,,,k}(x)}{x_i - x_j}$$

is the *k*th Lagrange polynomial that interpolates f at the *k + 1* points $x_0, x_1, ..., x_k$. We created polynomials using theorem 3.5 above as well as algorithm 3.1 in the book.

VALIDATION AND RESULTS

Using the data from Table 1 and Table 2, we used the methods above to generate interpolating polynomials to approximate specific heats for $x = 25°C$ and for $x = 85°C$. We then evaluated the accuracy of our approximations by computing the relative errors compared to the actual values of specific heat for $x = 25°C$ and for $x = 85°C$, which are 4180 J/(kg °C) and 4203 J/(kg °C) respectively. Below are results from our polynomials.

Results from Table 1 for x = 25°C

Method	Approximation (J/kg-C)	Relative Error
Cubic Spline	4180.542985	1.299007 * 10^-4
Lagrange	4179.441592	1.335904 * 10^-4
Neville's Method	4179.441592	1.335904 * 10^-4

Results from Table 2 for x = 85°C

Method	Approximation (J/kg-C)	Relative Error
Cubic Spline	4203.257565	6.128123 * 10^-5
Lagrange	4202.234375	1.821616 * 10^-4
Neville's Method	4202.234375	1.821616 * 10^-4

For the initial temperatures, 25°C and 85°C, the Cubic Spline method approximated results closest to the actual specific heat of these temperatures with the lowest relative errors as expected. For $x = 25$°C, the approximation from the Cubic Spline polynomial is 4180.542985 J/(kg °C) which yielded the smallest relative error of $1.299007 * 10^{-}4$. For $x = 85$°C, the approximation from the Cubic Spline polynomial is 4203.257565 J/(kg °C) which again yielded the smallest relative error of $6.128123 * 10^{-}5$. Both of the approximations from the Lagrange and Neville's polynomial's yielded the same specific heats. This was expected as the Neville's polynomial is constructed using lower-degree Lagrange polynomials.

Visual Representation

Below are plots created from the constructed polynomials, with Specific Heat J/(kg °C) and Temperature(°C) on the X and Y axis's respectively.

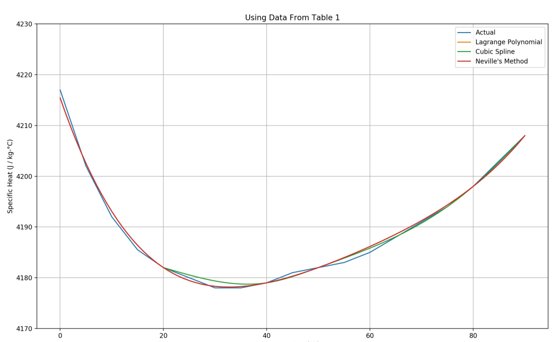

Figure 1. Using Data from Table 1

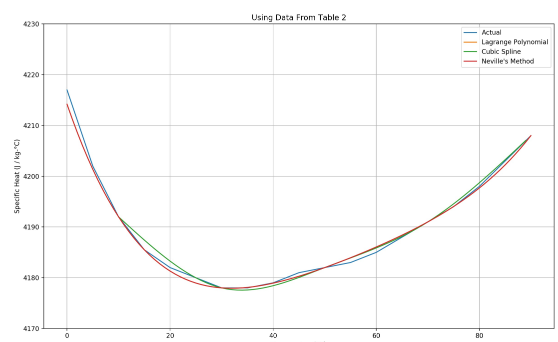

Figure 2. Using Data from Table 2

From Figure 1 and Figure 2 we can see how accurate our polynomials are compared the true line of temperatures. With the Cubic Spline polynomial hugging the actual line the closest, we can see visually how our hypothesis is justified. In addition, Neville's and Lagrange share the same line, supporting our results from Table 1 and Table 2.

CONCLUSION

There are important considerations that have an impact on the numerical methods and approaches we used for carrying out validation and prediction. The accuracy of our model's solution relative to that of the actual mathematical model showed that our method was successful at predicting the specific heat of water at our chosen temperatures. Our computational model had high accuracy relative to that of the true, physical system of specific heat. In summary, validation is a process which involves measurements and computational modeling for assessing how well a model represents reality.

This project allowed us to connect the theory and foundations of numerical methods with their practical applications in many different fields and domains.

Topic Applications

The specific heat of water is five times that of glass and ten times that of iron, which means that it takes five times as much heat to raise the temperature of water the same amount as for glass and ten times as much heat to raise the temperature of water as for iron. In fact, water has one of the largest specific heats of any material, which is important for sustaining life on Earth, as water covers about 70% percent of Earth's surface. Waters high specific heat plays a vital role because it is able to absorb a lot of heat without a significant rise in the temperature. Our project demonstrates a real world application of utilizing approximation methods. Our approximation methods could be of great use for practitioners dealing with water. This is relevant in today's age, where climate change is a critical issue. These methods could potentially help scientists understand the direct implications of changing temperatures and how they affects our oceans.

REFERENCES

1. Lumen Learning: Boundless Physics, Heat and Heat Transfer: Specific Heat, *https://courses.lumenlearning.com/boundless-physics/chapter/specific-heat/* (accessed Dec 2019)
2. Burden, Richard L, et al. *Numerical Analysis*. 10th ed., Cengage Learning, 2015.