CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING

Term Paper by BYFORD

A HIGH ORDER MULTIVARIATE APPROXIMATION SCHEME FOR SCATTERED DATA SETS

Date: 20-11-22

Contributed By:

Aryan Kushwaha : 2021CH10408 **Sameer Vidyarthi : 2021CH10423**

Akshay Malhotra: 2021CH10380 **Divy Kairoth : 2021CH10021**

CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING

Multivariate Approximation Scheme for Scattered Data Sets

ABSTRACT:

Here, depending on whether measurement error is included in the data points or not, we propose a method for roughly converting a given scattered data set into a target function that may be used as either an interpolation or a regression. We use each data point as a Taylor series in this method to obtain a function at a general d-dimensional location, where high order derivatives act as random variables. The estimation coefficients are obtained by solving an equality restricted least squares problem and minimising the error function at each position. Furthermore, by including more data points that provide the gradient, the accuracy of our approximation can be improved. Numerical study suggests that the advantage of this high order approximation is its great accuracy, even

Link of research paper we followed-
https://www.sciencedirect.com/science/article/pii/S002199911000241X

Introduction:

Since antiquity, the problem of approximating a continuous target function for finite measurements has been studied. Regression and interpolation techniques are used extensively in computing. It has a tangential application in the numerical method for solving differential equations since discrete data may not be sufficient and a continuous function may be required. Spectral approaches might make use of polynomial interpolation. The meshless technique for differential equations supports the use of -splines and radial basis function approximation. In order to sample their data, we may as well use it to create surrogate response functions. This class of approximated functions has applications across a wide range of scientific and technical disciplines whenever it is necessary to monitor the variability of a quantity with respect to any aspect in order to make predictions about the future or for other purposes.Borchardt and Kronecker conducted the initial research on multivariable interpolation. For non-uniform grid multivariate polynomial interpolation, de Boor and Ron presented the "least solution." The study of multivariate rational interpolation is in. Numerous approaches are developed expressly for interpolating scattered data sets in multi-dimensional domains, in addition to expansions of one-dimensional interpolation methods. Among the most widely used techniques for multivariate interpolation are Shepard's method, Kriging, and radial basis function interpolation. You can find in-depth analyses of dispersed data interpolation techniques.

This study was inspired by earlier work on creating a comparable uni-variable interpolation and regression system for arbitrary grids, which showed that it converged more quickly than polynomial order (exponential order). We now broaden our research to include a multidimensional framework for dispersed data points. It can also be shown that more "gradient data points" can be used to increase our system's precision. To achieve the approximation, our method employs a strategy of minimising the error at each particular position. Since we are estimating separately at each point, we are unable to establish a class of function for our approximation, and our method can be both a highly converging interpolation and a regression scheme.

CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING

The further of this paper has been divided into sections:

Section 2:Lays the mathematical foundation for our scheme.

Section 3:We show how our scheme is both an interpolation and regression based on presence of measurement errors.

Section 4:Describes the four controlling parameters that can be used to make our approximation more accurate with examples depicting variation of our approximation function with these parameters.

Section 5:It contains the scheme for calculating these parameters with the help of the data set provided

Section 6:It contains example of of our approximation scheme with its parameters calculated numerically with our algorithm and depicting the comparative accuracy and robustness of our approximations. Concluding remarks ,self assessment.

Section 2:

Mathematical Basis:

f=target function

We consider a target function f in a d-dimensional space. The values of the function is measured at n_v points x_{vi}, $i = 1,\ldots,n_v$. Each x_{vi} is a d-dimensional vector $x_{vi} = (x_{vi1},\ldots,x_{vid})$, representing a point in the d-dimensional space. These points are called "value nodes". The measurement of the function at x_{vi} is $\hat{f}(x_{vi})$, which is $f(x_{vi})$ plus a measurement error. The estimated size of the measurement error $\sigma_{vi} \geqslant 0$ is assumed known. When $\sigma_{vi} = 0$, we assume that the measurement is exact, and $\hat{f}(x_{vi}) = f(x_{vi})$. The data points $(x_{vi}, \hat{f}(x_{vi}))$, $i = 1,\ldots,n_v$ are called "value data points".

The gradient of the function is measured at n_g points x_{gi}, $i = 1,\ldots,n_g$. Each x_{gi} is also a d-dimensional vector $x_{gi} = (x_{gi1},\ldots,x_{gid})$, representing a point in the d-dimensional space. These points are called "gradient nodes". The measurement of the gradient of f at x_{gi} is a d-dimensional vector $\nabla\hat{f}(x_{gi}) = (\nabla_1\hat{f}(x_{gi}),\ldots,\nabla_d\hat{f}(x_{gi}))$, which is the exact gradient $\nabla f(x_{gi})$ plus a measurement error, where the exact gradient

$$\nabla f = (\nabla_1 f(x_{gi}),\ldots,\nabla_d f(x_{gi})), \quad \text{where } \nabla_k f(x_{gi}) = \frac{\partial f}{\partial x_k}(x_{gi}).$$

CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING

The estimated size of the measurement error $\sigma_{gi} \geqslant 0$ is again known. When $\sigma_{gi} = 0$, we assume that the measurement of the gradient is exact, and $\nabla\hat{f}(x_{gi}) = \nabla f(x_{gi})$. The data points $\langle x_{gi}, \nabla\hat{f}(x_{gi})\rangle,\ i = 1, \ldots, n_g$ are called "gradient data points".

The high order derivatives of f are represented using multi-index notation. Let $\kappa = (\kappa_1, \ldots, \kappa_d)$ be a d-dimensional multi-index, the κ-order derivative of f is defined as

$$f^{(\kappa)} = \frac{\partial^{|\kappa|} f}{\partial x_1^{\kappa_1} \cdots \partial x_d^{\kappa_d}}.$$

The absolute value, also called the total order of κ is defined as

$$|\kappa| = \sum_{k=1}^{d} \kappa_k,$$

and the factorial of κ is defined as

$$\kappa! = \prod_{k=1}^{d} \kappa_k!$$

In addition, the κ order multivariate monomial is defined as

$$x^{\kappa} = x_1^{\kappa_1} \cdots x_d^{\kappa_d}.$$

Specifically, we denote the unit multi-indices as e_k, $k = 1, \ldots, d$, which is defined as

$$e_k = (\delta_{k1}, \ldots, \delta_{kd}), \quad \text{where } \delta_{kj} = \begin{cases} 1, & k = j, \\ 0, & k \neq j. \end{cases}$$

These notations are used throughout this paper.

Using the information from the value data points and gradient data points, we construct an approximation function $\tilde{f}$. The specific form of the approximation is

$$\tilde{f}(x) = \sum_{i=1}^{n_v} a_{vi}(x)\hat{f}(x_{vi}) + \sum_{i=1}^{n_g} a_{gi}(x) \cdot \nabla\hat{f}(x_{gi}), \tag{1}$$

where each a_{vi} is a scalar valued function of x, and each a_{gi} is a vector valued function of x. Both a_{vi} and a_{gi} are called "basis functions". When considering a fixed x, we denote the values of the basis functions $a_{vi}(x)$ and $a_{gi}(x)$ as a_{vi} and a_{gi}. We also use the notations $a_v = (a_{v1}, \ldots, a_{vn_v})$, $a_g = (a_{g1}, \ldots, a_{gn_g})$, and $a = (a_v, a_g)$ to denote the value of all the basis functions at x. Once the basis functions are determined, the approximation function $\tilde{f}(x)$ is determined by Eq. (1).

In the rest of Section 2, we discuss our approach of constructing these basis functions. They are constructed so that the approximation error $\tilde{f}(x) - f(x)$ is small for infinitely differentiable functions. Section 2.3 transforms this objective into a least squares problem, which can be solve to obtain the values of the basis functions.

Representing the approximation error with Taylor expansions

Incorporating Eq. (1) into the approximation error at some point x in the d-dimensional space, we get

$$\begin{aligned}\tilde{f}(x) - f(x) &= \sum_{i=1}^{n_v} a_{vi}\hat{f}(x_{vi}) + \sum_{i=1}^{n_g} a_{gi}\cdot\nabla\hat{f}(x_{gi}) - f(x)\\ &= \sum_{i=1}^{n_v} a_{vi}f(x_{vi}) + \sum_{i=1}^{n_g} a_{gi}\cdot\nabla f(x_{gi}) - f(x) + \sum_{i=1}^{n_v} a_{vi}(\hat{f}(x_{vi}) - f(x_{vi})) + \sum_{i=1}^{n_g} a_{gi}\cdot(\nabla\hat{f}(x_{gi}) - \nabla f(x_{gi})).\end{aligned} \tag{2}$$

Since we consider the approximation error at a single point x, here we can denote the values of the basis function a_{vi} and a_{gi} at x as a_{vi} and a_{gi} without confusion. This notation will be used throughout the paper.

In order to estimate this approximation error, we expand each $f(x_{vi})$ around x using the multivariate Taylor's theorem:

$$f(x_{vi}) = f(x) + \sum_{0<|\kappa|\le N}\frac{f^{(\kappa)}(x)}{\kappa!}(x_{vi}-x)^{\kappa} + \sum_{|\kappa|=N+1}\frac{f^{(\kappa)}(\xi_{i\kappa})}{\kappa!}(x_{vi}-x)^{\kappa}, \tag{3}$$

where κ is a multi-index notation defined in Section 2.1. The residual term of the Taylor expansion includes the high order derivatives of f at some unknown points $\xi_{i\kappa}$, which is generally different for each i and κ.

The order of the Taylor expansion is chosen so that the number of derivative terms in the expansion matches the degree of freedom provided by the data points. More precisely, N is chosen so that

$$|\{\kappa : |\kappa| < N\}| \approx n_v + dn_g. \tag{4}$$

This choice of Taylor expansion order is determined by the maximum amount of information that can be extracted from the given set of data. Consider a very smooth function whose high order derivatives decay very fast. With n_v value data points and n_g gradient data points, a total of $n_v + dn_g$ number of lowest order function derivatives $f^{(\kappa)}(x)$ can be uniquely determined. This yields an accurate approximation given that higher order derivatives are relatively unimportant. When the function to be approximated is less smooth, data points that are far away is not useful in determining the derivatives of the function. Therefore, the amount of high order information that can be extracted from the data is always bounded by $n_v + dn_g$. For this reason, we always limit the order of our Taylor expansion, so that it contains approximately $n_v + dn_g$ terms.

Similarly, each $\nabla f(x_{gi}) = f^{(e_k)}(x_{gi})$ can also be expanded around x with Taylor's theorem:

CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING

$$\nabla_k f(x_{gi}) = \sum_{|\kappa|<N} \frac{f^{(\kappa+e_k)}(x)}{\kappa!}(x_{gi}-x)^{\kappa} + \sum_{|\kappa|=N} \frac{f^{(\kappa+e_k)}(\eta_{ik\kappa})}{\kappa!}(x_{gi}-x)^{\kappa}$$
$$= \sum_{\substack{0<|\kappa|\leqslant N \\ \kappa_k>0}} \frac{f^{(\kappa)}(x)}{(\kappa-e_k)!}(x_{gi}-x)^{\kappa-e_k} + \sum_{\substack{|\kappa|=N+1 \\ \kappa_k>0}} \frac{f^{(\kappa)}(\eta_{ik\kappa})}{(\kappa-e_k)!}(x_{gi}-x)^{\kappa-e_k}, \tag{5}$$

where e_k is the unit multi-index defined in Section 2.1. The residual term of the Taylor expansion includes the high order derivatives of f at some unknown points $\eta_{ik\kappa}$, which is generally different for each i, k and κ.

Incorporating these Taylor expansions into Eq. (2), the error of the approximation can be represented as

$$\tilde{f}(x) - f(x) = \sum_{|\kappa|\leqslant N} f^{(\kappa)}(x)\mathcal{X}_\kappa(x;a) + \sum_{|\kappa|=N+1} \sum_{i=1}^{n_v} f^{(\kappa)}(\xi_{i\kappa})\left(a_{vi}\frac{(x_{vi}-x)^{\kappa}}{\kappa!}\right) + \sum_{|\kappa|=N+1} \sum_{\substack{k=1 \\ \kappa_k>0}}^{d} \sum_{i=1}^{n_g} f^{(\kappa)}(\eta_{ik\kappa})\left(a_{gik}\frac{(x_{gi}-x)^{\kappa-e_k}}{(\kappa-e_k)!}\right)$$
$$+ \sum_{i=1}^{n_v}(\hat{f}(x_{vi}) - f(x_{vi}))a_{vi} + \sum_{k=1}^{d}\sum_{i=1}^{n_g}(\nabla_k\hat{f}(x_{gi}) - \nabla_k f(x_{gi}))a_{gik}, \tag{6}$$

where

$$\mathcal{X}_\kappa(x;a) = \begin{cases} \sum_{i=1}^{n_v} a_{vi} - 1, & |\kappa| = 0, \\ \sum_{i=1}^{n_v} a_{vi}\frac{(x_{vi}-x)^{\kappa}}{\kappa!} + \sum_{\substack{k=1 \\ \kappa_k>0}}^{d}\sum_{i=1}^{n_v} a_{gik}\frac{(x_{gi}-x)^{\kappa-e_k}}{(\kappa-e_k)!}, & |\kappa| > 0. \end{cases} \tag{7}$$

Eq. (6) splits the approximation error $\tilde{f}(x) - f(x)$ into three distinct parts:

1. The first line represents the contributions from the derivatives of the f at x, i.e.,

   $f^{(\kappa)}(x), \quad 0 \leqslant |\kappa| \leqslant N.$

These quantities are unknown; however, their coefficients in Eq. (6), $\mathcal{X}_\kappa(x;a)$, are functions of x and the values of the basis functions a_{vi} and a_{gi} at x.

2. The second and third lines are the residuals of the Taylor expansions. They represent the contribution from the high order derivatives of f at unknown points $\xi_{i\kappa}$ and $\eta_{ik\kappa}$, i.e.,

   $f^{(\kappa)}(\xi_{i\kappa}), \ i = 1,\ldots,n_v \quad \text{and} \quad f^{(\kappa)}(\eta_{ik\kappa}), \ i = 1,\ldots,n_g, \ k = 1,\ldots,d$

for all $|\kappa| = N + 1$. These quantities are unknown; however, their coefficients in Eq. (6),

$$\left(a_{vi}\frac{(x_{vi}-x)^{\kappa}}{\kappa!}\right) \quad \text{and} \quad \left(a_{gik}\frac{(x_{gi}-x)^{\kappa-e_k}}{(\kappa-e_k)!}\right),$$

are functions of x and the values of the basis functions a_{vi} and a_{gi} at x.

3. The last line represents the contribution from the measurement errors

   $\hat{f}(x_{vi}) - f(x_{vi}) \quad \text{and} \quad \nabla_k\hat{f}(x_{gi}) - \nabla_k f(x_{gi}).$

These measurement errors are unknown; however, their coefficients in Eq. (6) are simply the values of the basis functions a_{vi} and a_{gi} at x. When measurement errors are absent, this part is absent from Eq. (6).

Least square for the approximation error:

CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING

We constrain the basis functions with

$$\sum_{i=1}^{n_v} a_{vi} \equiv 1, \tag{8}$$

so that the approximation (1) is exact for a constant function f. With this constraint, the formula for the approximation error (6) becomes

$$\tilde{f}(x) - f(x) = \sum_{0<|\kappa|\leqslant N} f^{(\kappa)}(x)\mathcal{X}_\kappa(x;a) + \sum_{|\kappa|=N+1} \sum_{i=1}^{n_v} f^{(\kappa)}(\xi_{i\kappa}) \left(a_{vi} \frac{(x_{vi}-x)^\kappa}{\kappa!} \right)$$
$$+ \sum_{|\kappa|=N+1} \sum_{\substack{k=1 \\ \kappa_k>0}}^{d} \sum_{i=1}^{n_g} f^{(\kappa)}(\eta_{ik\kappa}) \left(a_{gik} \frac{(x_{gi}-x)^{\kappa-e_k}}{(\kappa-e_k)!} \right) + \sum_{i=1}^{n_v} (\hat{f}(x_{vi}) - f(x_{vi}))a_{vi} + \sum_{k=1}^{d} \sum_{i=1}^{n_g} (\nabla_k \hat{f}(x_{gi}) - \nabla_k f(x_{gi}))a_{gik}. \tag{9}$$

This equation represents the approximation error at each point as a linear combination of the following unknowns: the derivatives $f^{(\kappa)}(x)$, $f^{(\kappa)}(\xi_{i\kappa})$, $f^{(\kappa)}(\eta_{ik\kappa})$, and the measurement errors $\hat{f}(x_{vi}) - f(x_{vi})$ and $\nabla_k \hat{f}(x_{gi}) - \nabla_k f(x_{gi})$. We construct the following weighted L_2 norm of this linear combination:

$$\mathcal{Q}(x;a) = \sum_{0<|\kappa|\leqslant N} w_{|\kappa|}^2 \mathcal{X}_\kappa^2(x;a) + \sum_{|\kappa|=N+1} \sum_{i=1}^{n_v} w_{|\kappa|}^2 \left(a_{vi} \frac{(x_{vi}-x)^\kappa}{\kappa!} \right)^2 + \sum_{|\kappa|=N+1} \sum_{\substack{k=1 \\ \kappa_k>0}}^{d} \sum_{i=1}^{n_g} w_{|\kappa|}^2 \left(a_{gik} \frac{(x_{gi}-x)^{\kappa-e_k}}{(\kappa-e_k)!} \right)^2$$
$$+ \sum_{i=1}^{n_v} \sigma_{vi}^2 a_{vi}^2 + \sum_{k=1}^{d} \sum_{i=1}^{n_g} \sigma_{gi}^2 a_{gik}^2, \tag{10}$$

where the weights are

$$w_j = \beta\gamma^j, \quad j = 0, \ldots, N+1. \tag{11}$$

The purpose of the weighted L_2 norm $\mathcal{Q}(x;a)$ defined in Eq. (10) is to reflect the size of the approximation error in Eq. (9). The coefficients of the κ order derivatives in Eq. (9) are weighted by $w_{|\kappa|}$ in Eq. (10); and the coefficients of the measurement errors in Eq. (9) are weighted by their estimated sizes σ_{vi} and σ_{gi} in Eq. (10). The parameters "magnitude" β and "wavenumber" γ control the weights w_j. They are two important parameters of our scheme. For $\mathcal{Q}(\S)$ to best reflect the size of $\tilde{f}(x) - f(x)$, these should be chosen so that the weights $w_j = \beta\gamma^j$ reflect the size of $f^{(\kappa)}(x)$, $|\kappa| = j$. Sections 4.2 and 4.1 discuss how β and γ affect the behavior of the approximation $\tilde{f}(x)$. Section 5 shows our method of automatically calculating these parameters from the given data points.

The values of the basis functions at x is calculated by solving an equality constrained least squares problem

$$\min \mathcal{Q}(x;a) \quad \text{s.t.} \quad \sum_{i=1}^{n_v} a_{vi} = 1, \tag{12}$$

where $\mathcal{Q}(x;a)$ is defined in Eq. (10). The value of the approximation $\tilde{f}(x)$ is then determined by Eq. (1).

The approximation scheme:

Get the value data points$(xvi, f(xvi))$ and gradient data points and their measurement errors.

Determine the value of wavelenth ϒ and magnitude β .

For each point x where the value of function is needed make a matrix for the constraints and least square and solve the equality constrained least squares (12) for the variables avi and $agik$.

Use Eqn (1) for the value of $f(x)$.

3)Nature of our approximation scheme:

In this section we can prove that our function is an interpolation in case the mesurements are error free.

$\sigma_{vi} = 0$ and $x = x_{vi}$,

On solving the equality constrained least squares (12)

CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING

$$a_{vi'} = \begin{cases} 1, & i' = i, \\ 0, & i' \neq i, \end{cases}$$
$$a_{gi'} = 0, \quad \forall i'. \qquad (13)$$

When we put eqn 13 and x=xvi anto eqn 7and 10 we get

$$\mathcal{X}_\kappa(x; a) = 0 \ \forall \kappa \quad \mathcal{Q}(x; a) = 0.$$

So the error is minimised

And we get from (1)

$$\tilde{f}(x) = \hat{f}(x_{vi}). \qquad (14)$$

Thus it is proved that it is an interpolating scheme.

Whereas when the error is non zero the function oscillates around th value behaving more like a regression scheme such that it is passing through true points.

In the fig.1 we can see the linear approximation with different values of approximation errors.

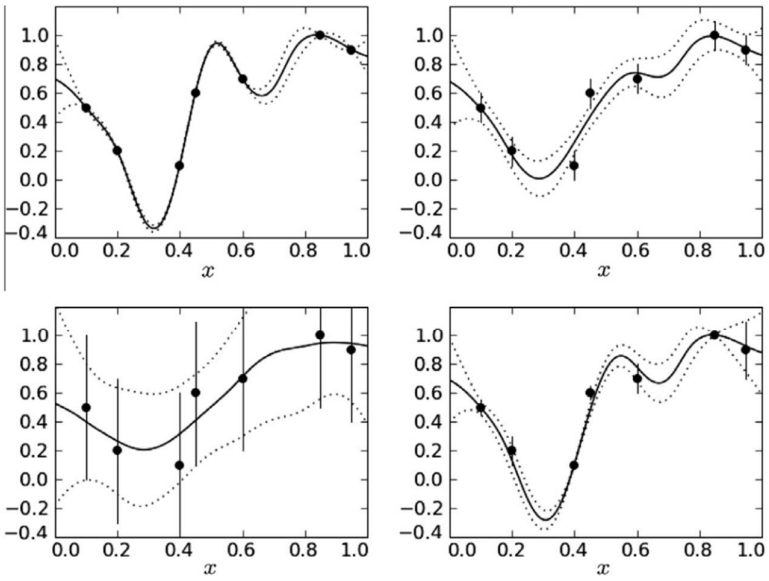


Fig. 1. $\tilde{f}$ calculated on the same data points with different σ_{vi}. $\beta = 0.5$, $\gamma = 10$. Filled circles represent the data points; the vertical bars indicates the size of σ_{vi}; the solid line is $\tilde{f}(x)$; the dotted lines represent the prediction interval $(\tilde{f}(x) - \sigma(x), \tilde{f}(x) + \sigma(x))$. $\sigma_{vi} = 0$ in the upper-left plot; $\sigma_{vi} = 0.1$ in the upper-right plot; $\sigma_{vi} = 0.5$ in the lower-left plot; in the lower-right plot, σ_{vi} has different values at each data point.

Prediction interval:

$$(\tilde{f}(x) - c\sigma(x), \tilde{f}(x) + c\sigma(x)), \quad \text{where } \sigma(x) = \sqrt{\mathcal{Q}(x; a)}.$$

1<c<3 higher value ensures a conventional interval

Confidance interval is plotted with dotted lines.

From the figure we can say that when the error is zero prediction interval size at data points is also zero

Size is more for extrapolation region than for intrapolation region.

Size increases as error increases

The prediction interval can thus serve as a powerful tool to find the value of wavelength.

It's easy to cite some complex mathematical equations which would require thorough transcription but our research paper's aim is to make it "comprehensible" even to a 12th standard student which has been briefed about the quantities mentioned in the paper and to actually dive into the topic chosen and add to it simultaneously.

SECTION 4 -

Getting down to business ,the two accessible entities which can be voluntarily changed in our estimation are defined below as γ and β :

$w_j = \beta * \gamma^j , j=0,\ldots,N+1. \quad -(11)$

The above formula is essential for understanding the content below.

4.1 Wavenumber *γ* :

In layman terms, this parameter extracts the change in weights w_j in Eq. (10) with respect to time.

(A)-- If $\gamma>1$:
w_j is directly proportional to j in the sense of increment. In this case, f˜(x) which is the interpolated function comes out to be less accurate which is not a good approximation. So, in essence, the higher more accurate terms in the interpolation formula are being neglected when *γ>1,* for instance this scheme can be demonstrated in a binomial approximation which states that (1 + x)^n where x is the variable is approximated to 1 + n*x or say the derivatives from second , third order and further in the taylor series are ignored which is a very rough approximation and this case reflects *γ be*ing greater than 1 in our scheme of things.
(B) – if $\gamma<1$:

Similarly, as explained in (A), every quantity here is just the contrary. So, w_j decreases with respect to increment in j. In this case, f˜(x) which is the interpolated function comes out to be more accurate which is actually a good approximation. So, in essence, the higher more accurate terms in the interpolation formula are not being neglected when *γ<1* but are used for approximating error analysis of f˜(x)-f(x). It is not that any derivatives are completely ignored but what changes is the more significant emphasis on the degree of derivatives be it the higher or lower ones accordingly.

Reading the above two subsections, the reader may seem that *γ<1* is easily more advantageous than *γ>1* but that's not really the case.
Here it gets a bit technical, the fact is that neither extremity is good if one wants the most accurate approximation implying *γ* should neither be too petty or large. However, one interesting assertion here is that *γ<<1(very petty)* is so much more dangerous than *γ>>1(very large)* for a particular set of coordinates on which interpolation is to be performed.
We discussed that a computer program which can predict an adequate value of *γ* for a set of data points can be of so much use. Further research in this field would likely enable us to build the computer program ourselves which would be a boon to the field and a personal achievement. Although, there are methods invented for appropriate selection of *γ* but even those methods are formulated to minimize the error so a newer more accurate method with less relative error could always be invented.

4.2. The magnitude *β(beta) :*

This entity comes into play only when errors are non-neglectable or else it acts like a harmless constant. In the former case, it gives a sense of how responsive the formulated function(f˜(x)) is and indicates the relative error.
Here, *β* having a large value is useful which enables more accurate interpolation which is uniform with most coordinates in the graph of the function and less "relative" errors(although its still considerably inaccurate with fluctuation in ordinates) however small value of *β* gives a non- uniform graph which is not accurate and not of much use.
Therefore, as was the case with *γ ,* selecting just the right *β* is salient for accuracy, essentially, for rational interpolation.

(Hint : appropriate selection of *β* lies in the definition of equation (11))

CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING

Through this multivariate data approximation model we aimed to use the beta and gama function

Mentioned in the paper to approx. the real life weather by taking data from online sources but we were only able to create our version of beta function ,since we were unable to create gama function from ourselves we were unable to reach the conclusion part of this paper.

So we started working on a covid outbreak model which uses differential eqn to predict

Outspread of virus and we also created a code to produce results and data graphs .

So our level 2 will consist of covid data model

Link for model we referred - COVID-19 SEIR Simulation (apmonitor.com)

CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING

Using SIER model to predict corona spread in the population

SIER-> (**SUSCEPTIBLE EXPOSED INFECTIOUS REMOVED)** Is a model used by many to predict the outbreak of virus , it consists of various parameters and different ODEs which are solved simultaneously to get the desired prediction , there are tons of model available on the net on seir to show the dynamics of corona outbreak some of them uses very complex partial differential equation which are out of scope for our subject , these models were very useful as by using them government authorities were able to take wise decision by carefully analysing which factor can reduce the spread and they were also able to predict peak time and slow downs in spread so that government could use it to impose necessary restrictions(like lock-down, work from home, 25% occupancy of offices)

The model we used in our term paper is relatively simple it consists

This model take mortality rate and birth rate to be zero, implying constant population N

Susceptible (s): population fraction that is susceptible to the virus
Exposed (e): population fraction is infected with the virus but does not transmit to others
Infectious (i): population fraction that is infected and can infect others
Recovered (r): population fraction recovered from infection and is immune from further infection

Differential equations of model to get results-

$$ds/dt = -(1-u)\beta si$$

$$de/dt = (1-u)\beta si - \alpha e$$

$$di/dt = \alpha e - \gamma i$$

$$dr/dt = \gamma i$$

CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING

meaning of parameters used above

α=1/(t_incubation)

γ=1/(t_infective)

β=R.* γ

u= it signifies proportion of people following proper social distancing norms , ranging from 0-1

t_incubation->it represent the time gap between the day person was infected to the day on which that person starts to show symptoms.

t_infective->it denotes the time in which an infected person can spread the virus to someone else.

R. ->it depends on the number of people an infected person can spread the virus.

While the author of this model used standard libraries present in python to calculate values . Instead we solved these differential equation using shooting method with explicit Euler method , we didn't use higher order methods the data became inconsistent with the results by author

t_incubation=5.1 days

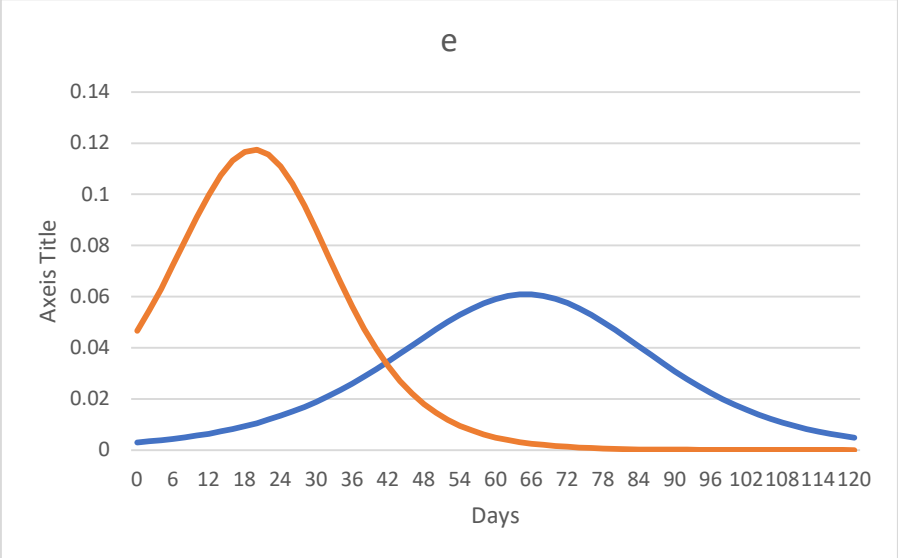
t_infective = 3.3 days

R. = 2.4 days

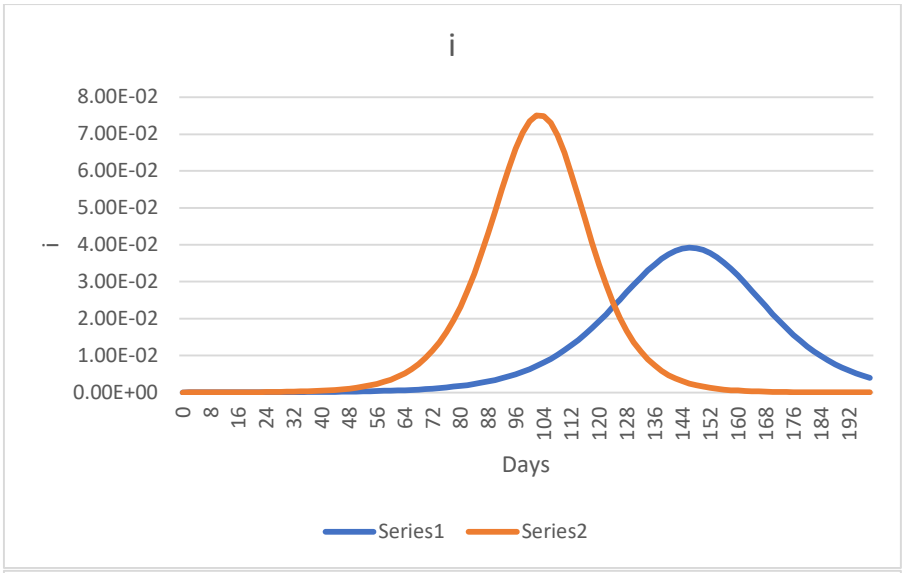
N=33517 (total population in model)

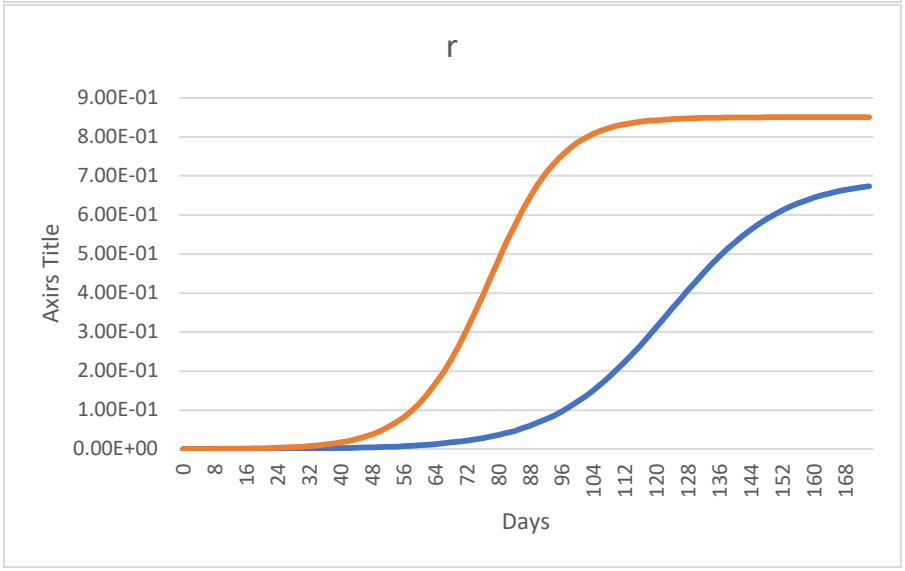
Graph plots from our model of function in cpp;

CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING



CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING





Blue line-> u=0.3

Orange line ->u=0.1

CLL 113: NUMERICAL METHODS IN CHEMICAL ENGINEERING

For reference_>

u=0 : no social distancing

u=0.1 : mask

u=0.2 : mask and hybrid classes

u=0.3 : mask and online classes

conclusion of graphs

1 As more social distancing is used Susceptible goes down slower.

2 Exposer(e) reches its maximum at longer time and it is smaller in magnitude

3 infection(i) also reaches its maximum at longer time and has a low peak

4 But apparently recovery is slower as well

Our function can be used to produce results by varying variables

The codes for finding this data and beta function are given in the file