USING THE CUBIC SPLINE INTERPOLATION METHOD TO APPROXIMATE SOME REAL DATA

Le Thi Thu[(1)]

(1)Thu Dau Mot University
Corresponding author: thult.khtn@tdmu.edu.vn

DOI: 10.37550/tdmu.EJS/2024.01.521

Article Info

Volume: 6
Issue: 01
March 2024
Received: Nov. 27th, 2023
Accepted: Feb. 25th, 2024
Page No: 93-101

Abstract

In engineering and science problems, the data being considered are often known in the form of sets of discrete points, not as a continuous function. In applications, the values of the discrete data at specific points may be required to estimate. To solve these problems, a common mathematical ap-proach is to use the interpolation method. In this paper, we use the cubic natural spline interpolation method to build approximate functions for some real data in Vietnam such as data on rice output and rice area of cultivation, data on expenditure and income per capita. The data used in this paper are extracted from the General Statistics Office of Vietnam. We develop some Matlab programs to find the coefficients of splines and calculate the values of the interpolation functions at specific points. Using the obtained interpolation function, some missing values in the data on income and expenditure per capita in the Southeast region of Vietnam in 2019 and 2021 are estimated. Further-more, some unknown data on rice productivity and cultivation area are obtained in a similar manner. A good agreement between the calculated values and actual values are found.

Keywords: cubic spline, interpolation, real data, statistics

1. Introduction

In mathematics, we often encounter problems related to investigating and calculating the value of a function $y = f(x)$. However, there are many cases in real life where we can not obtain an explicit expression of the function $f(x)$, but a set of discrete values y_i at the corresponding points $x_i \in [a, b]$ instead. From those data sets, we would like to estimate the value of the function $f(x)$ at any value $x \in [a, b]$. To solve this problem, a common mathematical approach is to use the interpolation method (Anh, 1998; Hoffman and Frankel, 2001; Richard and Douglas, 2011). The interpolation method helps us build an approximate function for the data set (x_i, y_i). Using the obtained interpolation function, we can easily compute the value of $f(x)$ at any $x \in [a, b]$.

There are several common methods for interpolating the data such as Newton interpolation and Lagrange interpolation. These are polynomial interpolation methods. Many studies successfully applied these polynomial interpolation approaches to solve problems in Agriculture and Health (Thuy and Ha, 2022), as well as in Education (Hussain, Srivastav, & Thota, 2014) or Animal husbandry (Çelik, 2018). However, the main disadvantage of the polynomial interpolation methods is that if there are many interpolation points, the degree of the interpolation polynomial will be very high, which is inconvenient in calculation (Anh, 1998). Furthermore, high-degree polynomials can oscillate unpredictably, that is, a minor fluctuation over a small portion of the interval can induce large fluctuations over the entire range (Richard and Douglas, 2011). One way to overcome this issue is using spline interpolation.

Splines can be of any degree. Linear splines are simply straight line segments connecting each pair of data points. They are independent of each other from interval to interval. Linear splines yield first-order approximating polynomials. The slopes (i.e., first derivatives) and curvature (i.e., second derivatives) are discontinuous at every data point. Quadratic splines yield second-order approximating polynomials. The slopes of the quadratic splines can be forced to be continuous at each data point, but the curvatures are still discontinuous (Hoffman and Frankel, 2001).

A cubic spline yields a third-degree polynomial connecting each pair of data points. The slopes and curvatures of the cubic splines can be forced to be continuous at each data point. These requirements are necessary to obtain the additional conditions required to fit a cubic polynomial to two data points. Higher-degree splines can be defined similarly. However, cubic splines have proven to be a good compromise between accuracy and complexity (Hoffman and Frankel, 2001). Consequently, the cubic spline interpolation is used in this study.

There have been articles applying the cubic spline interpolation to COVID-19-related data (AlAmeri, 2021; Munandar et al., 2022) or heat transfer problems (Chikwendu et al., 2015). However, to the extent of our knowledge, there are no studies using the cubic spline interpolation to estimate data on agriculture or data on people's income and expenditure in Vietnam.

In this paper, we use the cubic natural spline interpolation method to build approximate functions for data on rice output and rice area of cultivation in Vietnam, as well as data on expenditure and income per capita in the Southeast region of Vietnam. The data used in this paper are extracted from the General Statistics Office of Vietnam. To find the coefficients of splines as well as the value of the interpolation function at specific values, we developed some Matlab software. The software helps to reduce calculation time as well as errors in manual calculation, especially when the data is big.

The rest of this paper is organized as follows. In section 2, we present the data, the cubic natural spline interpolation method, as well as programs we programmed in Matlab to support finding splines. In section 3, we describe the results when applying the cubic natural spline interpolation method to the data we have collected. Finally, some conclusions are presented in section 4.

2. Materials and methods

2.1. Materials

In this papers, we use data on rice area of cultivation (unit: thousand hectares) and rice output (unit: thousand tons) in Vietnam from 2015 to 2022. This data is extracted from the data of the General Statistics Office of Vietnam and shown in Table 1.

TABLE 1. Rice area of cultivation and rice output in Vietnam from 2015 to 2022

Year	**2015**	**2016**	**2017**	**2018**	**2019**	**2020**	**2021**	**2022**
Area of cultivation (X)	7828	7737.1	7705.2	7570.9	7469.9	7278.9	7238.9	7109
Output (Y)	45091	43109	42738.9	44046	43495.4	42764.8	43852.6	42660.7

In addition, we also collect data on per capita income (unit: million VND/month) and per capita expenditure (unit: million VND/month) in the Southeast region of Vietnam. The data are presented in Table 2.

TABLE 2. Per capita income and expenditure in the Southeast region of Vietnam

Year	**2010**	**2012**	**2014**	**2016**	**2018**	**2019**	**2020**	**2021**	**2022**
Income (X)	2.304	3.173	4.125	4.662	5.709	6.280	6.025	5.794	6.334
Expenditure (Y)	1.640	2.036	2.282	2.846	3.149		3.719		3.456

This data is obtained from the website of the General Statistics Office of Vietnam. We can see that the statistical data on income in 2019 and 2022 are available, but those on expenditure are unknown.

2.2. The cubic natural spline interpolation method

In this section, we provide a brief discussion on the cubic natural spline interpolation method. Given a function $f(x)$ defined on $[a,b]$ and a set of nodes $\Delta = \{a = x_0 < x_1 < \cdots < x_n = b\}$. A natural cubic spline interpolant $S(x)$ for function $f(x)$ on $[a,b]$ is a function that satisfies the following conditions (Richard and Douglas, 2011):

(a) $S(x)$ is a cubic polynomial, denoted $S_j(x)$, on subinterval $[x_{j-1}; x_j]$ for each $j = 1,2,\dots,n$;
(b) $S_j(x_{j-1}) = f(x_{j-1})$ and $S_j(x_j) = f(x_j)$ for each $j = 1,2,\dots,n$;
(c) $S_j(x_j) = S_{j+1}(x_j)$ for each $j = 1,2,\dots,n-1$;
(d) $S'_j(x_j) = S'_{j+1}(x_j)$ for each $j = 1,2,\dots,n-1$;
(e) $S''_j(x_j) = S''_{j+1}(x_j)$ for each $j = 1,2,\dots,n-1$;
(f) $S''(a) = S''(b) = 0$ (natural or free boundary conditions)

The following shows how to find cubic natural splines (Anh, 1998).

Let

$$m_j = S''(x_j); h_j = x_j - x_{j-1}, j = \overline{0,n}. (1)$$

Firstly, we need to compute the following parameters:

$$\lambda_j = \frac{h_{j+1}}{h_j + h_{j+1}}, j = \overline{1, n-1}; (2)$$

$$\mu_j = 1 - \lambda_j = \frac{h_j}{h_j + h_{j+1}}, j = \overline{1, n-1}; (3)$$

$$d_j = \frac{6}{h_j + h_{j+1}}\left(\frac{y_{j+1} - y_j}{h_{j+1}} - \frac{y_j - y_{j-1}}{h_j}\right), j = \overline{1, n-1} (4)$$

Secondly, we need to solve a linear system of equations with following expanded matrix $(A \vee B)$ to find the values $m_0, m_1, \dots, m_n$

$$(A \vee B) = \left(\begin{array}{cccccccc|c} 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ \mu_1 & 2 & \lambda_1 & 0 & \cdots & 0 & 0 & 0 & d_1 \\ 0 & \mu_2 & 2 & \lambda_2 & 0 & \cdots & 0 & 0 & d_2 \\ & & & \cdots & & & & & \cdots \\ 0 & 0 & 0 & \cdots & 0 & \mu_{n-1} & 2 & \lambda_{n-1} & d_{n-1} \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 0 \end{array}\right) (5)$$

It is obvious that this matrix shows the system always has a unique solution.

Finally, with the obtained values $m_0, m_1, \dots, m_n$, we can find cubic natural splines on each $\nabla_j = [x_{j-1}; x_j]$ by following formula:

$$S_j(x) = \frac{m_{j-1}}{6h_j}(x_j - x)^3 + \frac{m_j}{6h_j}(x - x_{j-1})^3 + \frac{1}{h_j}\left(y_{j-1} - \frac{m_{j-1}h_j^2}{6}\right)(x_j - x)$$
$$+ \frac{1}{h_j}\left(y_j - \frac{m_j h_j^2}{6}\right)(x - x_{j-1}) (6)$$

Let us illustrate the natural cubic spline by applying it to the data points $(0,1), (1,e), (2,e^2), (3,e^3)$, which are the values of the function $f(x) = e^x$ for $x \in \{0,1,2,3\}$.

Applying (1), (2), (3), and (4) gives

$$h_1 = 1, h_2 = 1, h_3 = 1;$$
$$\lambda_1 = 0.5, \lambda_2 = 0.5;$$
$$\mu_1 = 0.5, \mu_2 = 0.5;$$
$$d_1 \approx 8.85747732603768, d_2 \approx 24.0771196613562$$

Solving the linear system of equations with following expanded matrix

$$\left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 0 \\ \mu_1 & 2 & \lambda_1 & 0 & d_1 \\ 0 & \mu_2 & 2 & \lambda_2 & d_2 \\ 0 & 0 & 0 & 1 & 0 \end{array}\right) = \left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 0 \\ 0.5 & 2 & 0.5 & 0 & 8.85747732603768 \\ 0 & 0.5 & 2 & 0.5 & 24.0771196613562 \\ 0 & 0 & 0 & 1 & 0 \end{array}\right)$$

we obtain $m_0 = 0, m_1 \approx 1.51370528570593, m_2 \approx 11.6601335092516, m_3 = 0$.

Applying (6), this leads to

$$S(x) = \begin{cases} 0.252284x^3 + 1.465998x + 1, x \in [0;1] \\ 1.691071x^3 - 4.316361x^2 + 5.782359x - 0.438787, x \in [1;2] \\ -1.943356x^3 + 17.4902x^2 - 37.830764x + 28.636628, x \in [2;3] \end{cases}$$

Figure 1 displays the graph of spline $S(x)$ and that of $f(x) = e^x$. In this figure, the red dashed line represents the graph of spline $S(x)$, and the blue line is the graph of the function $f(x)$. The red star marks in this graph correspond to the interpolation points. It can be seen that a good agreement between the graphs are found.

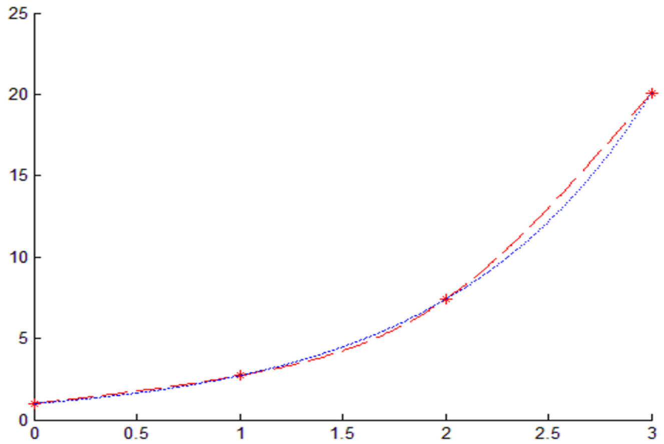

Figure 1. Graph of function $\mathrm{f(x) = e^x}$ and spline $\mathrm{S(x)}$. The red dashed line represents the graph of $\mathrm{S(x)}$, and the blue line is the graph of $\mathrm{f(x)}$. The red star marks correspond to the interpolation points.

2.3. Matlab programs to find cubic natural splines

It is clear that manual calculation to find cubic natural splines for big data is quite time-consuming and tedious, and often leads to mistakes in calculating. In this section, we present how to build some software in Matlab for this paper.

We build the first program to find cubic natural splines. The input data for this program are vector X and vector Y containing the values of $x_j, j = \overline{0,n}$ and $y_j, j = \overline{0,n}$, respectively. The returned result of the function in the first program is the matrix P containing the coefficients of spline $S_j(x)$ on each subinterval $\nabla_j = [x_{j-1}; x_j]$. The code of the first program is given as follows.

```
function P = CubicSpline(X, Y)
format longg;
n = length(X);
h = zeros(n-1,1);
z = zeros(n-1,1);
for i=1:(n-1)
 h(i) = X(i+1) - X(i);
 z(i) = (Y(i+1) - Y(i))/h(i);
end
mu = zeros(n-2,1);
lamda = zeros(n-2,1);
```

```
d = zeros(n-2,1);
for i=1:(n-2)
 mu(i) = h(i)/(h(i)+h(i+1));
 lamda(i) = 1-mu(i);
 d(i) = 6/(h(i)+h(i+1))*(z(i+1)-z(i));
end
B = zeros(n,1);
for i=2:(n-1)
 B(i) = d(i-1);
end
A = zeros(n);
A(1,1) = 1;
A(n,n) = 1;
for i=2:(n-1)
 A(i,i-1) = mu(i-1);
 A(i,i) = 2;
 A(i,i+1) = lamda(i-1);
end
m = inv(A)*B;
M = zeros(1, n-1);
N = zeros(1, n-1);
for i=1:(n-1)
 M(i) = 1/h(i)*(Y(i)-m(i)*h(i)^2/6);
 N(i) = 1/h(i)*(Y(i+1)-m(i+1)*h(i)^2/6);
end
syms x;
for i=1:(n-1)
 S = expand(m(i)/(6*h(i))*(X(i+1)-x)^3 + m(i+1)/(6*h(i))*(x-X(i))^3 + M(i)*(X(i+1)-x) + N(i)*(x-
X(i)));
 P(i,:) = sym2poly(S);
end
```

The second program is used to find the value of the spline function at specific points. The input data of this program are vectors X, Y, and Z. In which, X and Y are the vectors used in the first program to find splines. The vector Z contains the values at which we need to find the value of the spline function. The returned result of the second program is the values of spline at the corresponding values of vector Z. Below is the code of the second program.

```
function F = Valueofspline(X, Y, Z)
P = CubicSpline(X,Y);
n = length(X);
k = length(Z);
for i = 1:(n-1)
 for j = 1:k
 if Z(j)>X(i)
 if Z(j)<X(i+1)
 F(j) = polyval(P(i,:),Z(j));
 end
 end
 end
end
```

The final program helps us build the plot of the obtained splines. Below, we present the code of this program.

```
function PlotSpline(X, Y)
P = CubicSpline(X,Y);
n = length(X);
hold on
for i = 1:(n-1)
 xi = X(i):0.01:X(i+1);
 yi = polyval(P(i,:),xi);
 plot(xi,yi,'--b')
end
for j = 1:n
 plot(X(j), Y(j),'*r')
end
```

3. Results and discussions

3.1. Analysis on rice output and rice area of cultivation

In this section, we use the cubic natural spline interpolation method to find the approximate function for the data in Table 1 with the help of the first program presented in section 2.3. Then, we estimate rice outputs according to some values of rice cultivation area with the aid of the second program. In addition, the graph of the obtained spline is created by using the third program.

Notice that the rice area of cultivation in Vietnam from 2015 to 2022 in Table 1 has not been arranged in ascending order. So, firstly, we need to rearrange the data so that the values of vector X are in ascending order. Next, we find the cubic natural splines in each subinterval by calling the following commands.

```
>> X = [7109, 7238.9, 7278.9, 7469.9, 7570.9, 7705.2, 7737.1, 7828];
>> Y = [42660.7, 43852.6, 42764.8, 43495.4, 44046, 42738.9, 43109, 45091];
>> CubicSpline (X,Y)
```

The approximate function for the data in Table 1 is found in the following form

$$S(x) = \begin{cases} S_1(x), x \in [7109; 7238.9] \\ S_2(x), x \in [7238.9; 7278.9] \\ S_3(x), x \in [7278.9; 7469.9] \\ S_4(x), x \in [7469.9; 7570.9] \\ S_5(x), x \in [7570.9; 7705.2] \\ S_6(x), x \in [7705.2; 7737.1] \\ S_7(x), x \in [7737.1; 7828] \end{cases}$$

where $S_i(x) = a_{i1}x^3 + a_{i2}x^2 + a_{i3}x + a_{i4}, i = \overline{1,7}$, in which a_{ij} is the number in row i, column j of the following matrix:

$$\begin{bmatrix} -0.000899807302611782 & 19.1901903428015 & -136398.704269923 & 323146678.746981 \\ 0.00501399465457238 & -109.23807262078 & 793280.648497145 & -1920138610.1682 \\ -0.000519287471318058 & 11.5905491776519 & -86218.806711459 & 213790918.004441 \\ -0.000365827823363876 & 8.15156450489307 & -60529.9351044177 & 149826483.998629 \\ 0.00100429226344128 & -22.9675619906863 & 175069.859680964 & -444741011.448254 \\ -0.00161703715365488 & 37.6260402831416 & -291815.964559334 & 754408539.530529 \\ -0.000339221850062672 & 7.9662859268718 & -62335.4791294395 & 162570718.257316 \end{bmatrix}$$

The graph of the obtained spline $S(x)$ with the support of the third program is shown in Figure 2. The red star marks in this graph are the locations of points (x_i, y_i) of the data set. It is clear that the spline function always goes through the interpolation points and is smooth at the connection points.

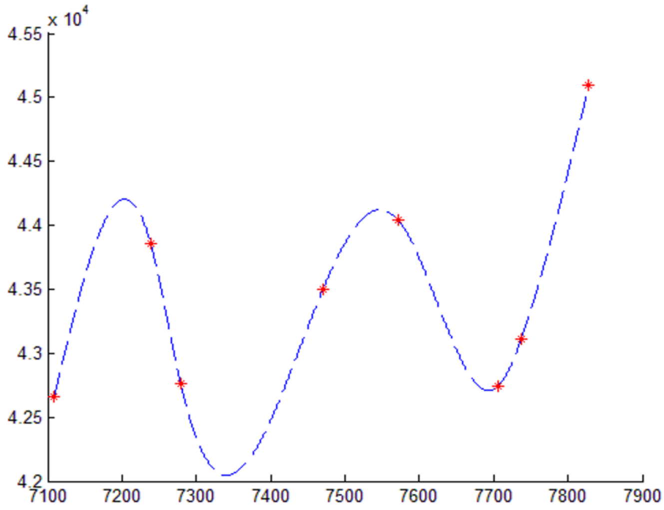

Figure 2. Graph of the cubic natural spline function for data on rice output and rice area of cultivation. The red star marks correspond to the interpolation points.

From the obtained spline interpolation function, we can estimate the rice outputs corresponding to the values of cultivation area in the interval [7109; 7828]. Here, we calculate the rice outputs corresponding to the cultivation areas 7150, 7250, 7350, 7550, 7650, 7750. We use the second program presented in section 2.3 by calling the following commands.

```
>> X = [7109, 7238.9, 7278.9, 7469.9, 7570.9, 7705.2, 7737.1, 7828];
>> Y = [42660.7, 43852.6, 42764.8, 43495.4, 44046, 42738.9, 43109, 45091];
>> Z =[7150, 7250, 7350, 7550, 7650, 7750];
>> Valueofspline(X, Y, Z)
```

The results of above commands after rounding to 1 decimal place are shown in Table 3.

TABLE 3. Estimations of rice outputs according to rice cultivation areas using spline interpolation results

Area of cultivation (Z)	7150	7250	7350	7550	7650	7750
Output	43597.4	43581.0	42057.1	44120.9	43027.5	43332.6

From the approximate data in Table 3, we found that the average rice productivity is 5.810213 tons/ha. And, from the actual data in Table 1, the average rice productivity is 5.801978 tons/ha. Thus, the difference in average rice productivity between real data and data estimated from the spline interpolation function is only 0.008234 tons/ha. This number shows that natural spline interpolation is a good method to approximate the data in Table 1.

3.2. Analysis on expenditure and income per capita

In this section, we use the cubic natural spline interpolation method to find the approximate function for the data in Table 2 with support of the first program presented in section 2.3. Then, we estimate expenditure per capita in 2019 and 2021 with support of the second program. In addition, the graph of the obtained spline is shown by using the third program.

Because in 2019 and 2022 there is statistical data on income, but no statistics on expenditure, so we find the cubic natural splines in each subinterval of data of the years 2010, 2012, 2014, 2016, 2018, 2020, and 2022 by calling the following commands.

```
>> X = [2.304, 3.173, 4.125, 4.662, 5.709, 6.025, 6.334];
>> Y = [1.640, 2.036, 2.282, 2.846, 3.149, 3.719, 3.456];
>> CubicSpline(X,Y)
```

We found the approximate function for the data in Table 2 of the following form

$$S(x) = \begin{cases} S_1(x), x \in [2.304; 3.173] \\ S_2(x), x \in [3.173; 4.125] \\ S_3(x), x \in [4.125; 4.662] \\ S_4(x), x \in [4.662; 5.709] \\ S_5(x), x \in [5.709; 6.025] \\ S_6(x), x \in [6.025; 6.334] \end{cases}$$

where $S_i(x) = a_{i1}x^3 + a_{i2}x^2 + a_{i3}x + a_{i4}, i = \overline{1,6}$, in which a_{ij} is the number in row i, column j of the following matrix:

$$\begin{bmatrix} -0.195344412694953 & 1.35022058054751 & -2.50769553301469 & 2.63937548586071 \\ 0.642784748110905 & -6.62793090116344 & 22.8069791184542 & -24.1351120705095 \\ -2.08333539500916 & 27.1078058699474 & -116.352935062378 & 167.209769928135 \\ 1.69076406294489 & -25.676749148998 & 129.728660435946 & -215.20102947626 \\ -11.0572022937302 & 192.657670641777 & -1116.74254214959 & 2156.83366904401 \\ 7.76835039632917 & -147.614194231047 & 933.395443709175 & -1960.526785889 \end{bmatrix}$$

The graph of the obtained spline $S(x)$ with the support of the third program is shown in Figure 3. The red star marks in this graph are the locations of points (x_i, y_i) of the data set. We can see that the spline function always goes through the interpolation points and is smooth at the connection points.

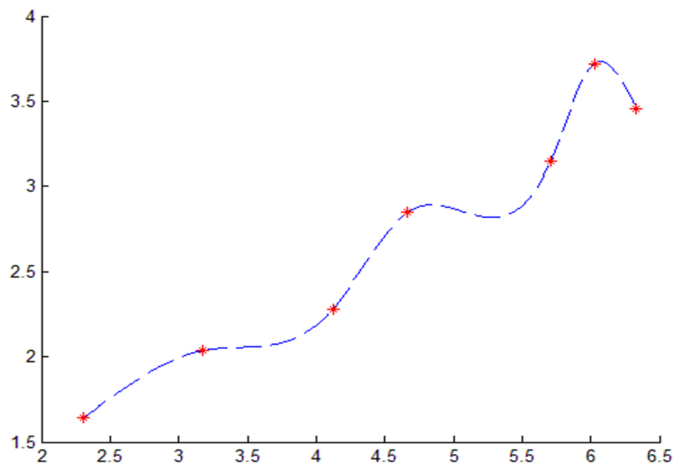

Figure 3. Graph of the cubic natural spline function for data on expenditure and income. The red star marks correspond to the interpolation points.

From the obtained spline interpolation function, we can estimate the expenditure per capita based on the income per capita in the range [2,304; 6.334]. In this part, we estimate the missing values in Table 2 for 2019 and 2021. According to the data in Table 2, the income per capita in the Southeast of Vietnam in 2019 was 6,280 million/month. Since $6.280 \in [6.025; 6.334]$ so we replace $x = 6.280$ into spline $S_6(x)$, we estimate that the expenditure per capita in 2019 is about 3,541 million/month. Similarly, the income per capita in 2021 was 5,794 million/month, we found the expenditure per capita in 2021 is about 3,325 million/month. These approximate values can be obtained using program 2 with the following commands.

```
>> X = [2.304, 3.173, 4.125, 4.662, 5.709, 6.025, 6.334];
>> Y = [1.640, 2.036, 2.282, 2.846, 3.149, 3.719, 3.456];
>> Z = [5.794, 6.280];
>> Valueofspline(X, Y, Z)
```

4. Conclusion

In this paper, we used the cubic natural spline interpolation method to find an approximate function for data on rice output and rice cultivation area in Vietnam from 2015 to 2022. From the obtained interpolation function, we have given estimations for rice output according to some values of rice cultivation area. We also compared the average rice productivity from real data and the approximate data obtained from the interpolation function. We found that the average rice productivity calculated from the approximate data is 5.810213 tons/ha, which is approximately the average value calculated from the real data (5.801978 tons/ha).

Further, we also applied the cubic natural spline interpolation method to build an approximate function for data on income and expenditure per capita in the Southeast region of Vietnam in 2010. 2012, 2014, 2016, 2018, 2020, and 2022. We used the obtained interpolation function to predict the missing values in the data in 2019 and 2021. The results show that expenditure per capita in the Southeast region in 2019 is 3,541 million/month, and in 2021 it is 3,325 million/month.

In addition, we also used Matlab software to build programs to find splines in each subinterval, as well as calculate the values of an interpolation function at specific values. These programs can be used to solve similar problems for any data.

References

AlAmeri, J. M. (2021). Cubic Spline Interpolation for Data Infections of COVID-19 Pandemic in Iraq. *Al-Qadisiyah Journal of Pure Science*, *26*(5). https://qu.edu.iq/journalsc/index.php/JOPS/article/view/1443

Anh, P. K. (1998). *Numerical analysis* (4th ed.). Hanoi National University Publishing House.

Çelik, Ş. (2018). Using Lagrange interpolation to determine the milk production amount by the number of milked animals. *ResearchGate*. https://www.researchgate.net/publication/327561652

Chikwendu, C. R., Oduwole, K., & Okoro, S. I. (2015). An application of spline and piecewise interpolation to heat transfer (Cubic case). *ResearchGate*. https://www.researchgate.net/publication/332843433_An_Application_of_Spline_and_Piecewise_Interpolation_to_Heat_Transfer_Cubic_Case

General Statistics Office (n.d.). gso.gov.vn. Retrieved November 6, 2023, from https://www.gso.gov.vn/

Hoffman, J. D., & Frankel, S. (2001). *Numerical Methods for Engineers and Scientists*. CRC Press.

Hussain, S., Srivastav, V., & Thota, S. (2014). Assessment of interpolation methods for solving the real life problem. *ResearchGate*. https://www.researchgate.net/publication/271520122

MATLAB. (n.d.). Retrieved November 6, 2023, from https://www.mathworks.com/products/matlab.html

Munandar, D., Suryaningrat, W., & Purwani, S. (2022). Polynomial interpolation and cubic spline to determine approximate function of COVID-19 tweet dataset. *ResearchGate*. https://www.researchgate.net/publication/361218928_Polynomial_Interpolation_and_Cubic_Spline_to_Determine_Approximate_Function_of_COVID-19_Tweet_Dataset

Richard L. Burden, & J. Douglas Faires (2011). *Numerical Analysis* (9th ed.). Cengage Learning.

Thuy, N. T., & Ha, L. T. (2022). Some applications of Lagrange interpolation polynomials in practice. *HNUE JOURNAL OF SCIENCE*, *67*(1), 10–18. https://doi.org/10.18173/2354-1059.2022-0002