IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 7 Issue 11, November 2020
ISSN (Online) 2348 – 7968 | Impact Factor (2020) – 6.72
www.ijiset.com

Defining the Optimal Number of Knots In B-Spline Using Sturges Rule

IGNM Jaya[1*] and Neneng Sunengsih[2]

[1,2]Department of Statistics, Universitas Padjadjaran, Bandung, Indonesia

*mindra@unpad.ac.id

Abstract

B-spline is a popular technique for nonlinear regression modeling. It has been used to model several phenomena with nonlinear relationships between independent and dependent variables. However, defining the optimal number of knots could be an issue. Too many knots lead to an overestimate problem. Inversely, with a too-small number of knots, the regression line could be over smooth. We propose to use Sturges rule to define the optimal number of bins, which is related to the optimal number of knots. We add smoothing parameter using ridge approach to avoid the over smoothing problem. We found this approach quite good with efficient computation time based on three different patterns of relationship between covariates and dependent variables.

Keywords: Basis functions, B-spline, Regression, Sturges rule

Introduction

In big data era, the data driven approach are widely used to model the relationship between independents and dependent variables ([1]; [2]; [3]). Nonparametric method becomes a popular technique in regression analysis to approximate the unknown *smooth functions* [4]. A common approach that is commonly used is spline technique. A ‘spline’ is a function that is established piecewise from polynomial functions [5]. B-spline is a class of spline model that is widely used ([5]; [6]). A B-spline curve is constructed based on linear combination of control points. It is a generalization of the Bezier curve A B-spline function is a combination of piecewise polynomial degree $r > 0$ with p interior knots. Knots is defined as a place where the piecewise polynomials are meet. Although it is used very often, B-spline has a sensitive problem, namely in determining the number of knots. Too many knots lead to an overestimate problem. Inversely, with a too-small number of knots, the regression line could be over smooth. We propose to use Sturges rule to define the optimal number of bins, which is related to the optimal number of knots.

Method

B-spline

The B-spline curve of degree r is a parametric curve constructed of a linear combination of B-spline basis function $B_{i,r}(x)$ and defined as [5]:

$$B(x) = \sum_{i=0}^{p+r} \gamma_i B_{i-r,r}(x) \ ; x \in [t_0, t_{p+1}] \tag{E.1}$$

For each of knots $t_{i-r}, t_{i-(r-1)}, \dots, t_0, t_1, \dots, t_{p+1}, \dots, t_{p+r+1}$, the B-spline basis functions recursively define:

$$B_{i-r,0}(x) = \begin{cases} 1 & \text{if } t_{i-r} \le x < t_{i-r+1} \\ 0 & otherwise. \end{cases} \tag{E.2}$$

$$B_{i-r,r}(x) = \frac{x - t_{i-r}}{t_i - t_{i-r}} B_{i-r,r-1}(x) + \frac{t_{i+1} - x}{t_{i+1} - t_{i-r+1}} B_{i-r+1,r-1}(x)$$

with the first and the last of $p + 1$ knots have similar values with the minimum(x) and maximum(x) respectively.

Sturges rule

Next, we discuss about the B-spline problem. The number of knots and position, and selected polynomial degree will be common issues in B-spline. To avoid the problems, we propose to use fixed degree $r = 1$ and define the number of knots and its positions based on Sturges formula. A rule for determining the desirable number of groups into which a distribution of observations should be classified. The rule is [7]:

(a) Define the range R= maximum (x) – minimum (x)

(b) Define the number of groups or classes is K=1 + 3.3 log n, where n is the number of observations.

(c) Define the interval class Interval = R/K

(d) The knots is defined by $\zeta_i = \min(x) + (i - 1) * \text{Interval}; \text{i} = 1, \dots, \text{K}$

IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 7 Issue 11, November 2020
ISSN (Online) 2348 – 7968 | Impact Factor (2020) – 6.72
www.ijiset.com

To avoid overfitting or over flexible we add roughness penalty using ridge approach.

Result and discussion

To explain in detail the B-spline approach we present a simple example how to calculate the B-spline functions and construct the B-spline curve. The R-code from [5] is modify and used in all this study. Let assume $x \in [0,1]$ with $y = \sin(2\pi x)$ with number of observation $n = 100$

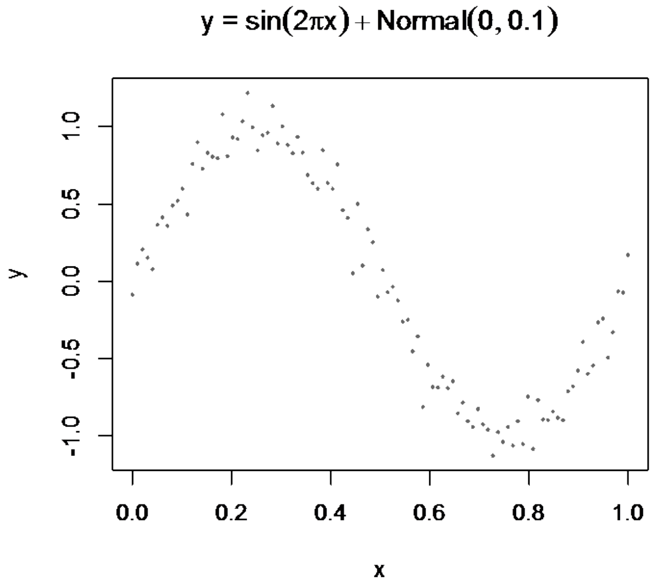

Figure 1. Observed data x versus y

First we explain the step by step compute the B-spline basis functions and obtain the fitted value. We introduce two different polynomial degree $r = \{1, 2\}$ with a single knot $p = 1$ with value $\zeta_1 = 0.5$.

B-spline with piecewise polynomial with degree $r = 1$ (linear)

First, we compute the matrix of the B-spline basis functions

$$\boldsymbol{Z} = \left[\boldsymbol{B}_{-r}^{r}(x), \boldsymbol{B}_{(1-r)}^{r}(x), \dots, \boldsymbol{B}_{p}^{r}(x)\right] = [\boldsymbol{B}_{-1}^{1}(x), \boldsymbol{B}_{0}^{1}(x), \boldsymbol{B}_{1}^{1}(x)].$$

The inner knot $\zeta_1 = 0.5$, boundary konts $\{\zeta_0 = 0, \zeta_2 = 1\}$ and the additional knots $\{\zeta_{-1} = \zeta_0 = 0, \zeta_2 = \zeta_2 = 1\}$ and the knot vector becomes (0, 0, 0.5, 1, 1).

B-spline basis function for $j = 0$, $B_{-1}^{1}(x)$ is:

$$B_{-1}^{1}(x) = \frac{x-\zeta_{-1}}{\zeta_0-\zeta_{-1}}B_{-1}^{0}(x) + \frac{\zeta_1 - x}{\zeta_1-\zeta_0}B_0^0(x) \tag{E.3}$$

where $B_{-1}^{0}(x) \equiv 0$ because $\zeta_{-1} = \zeta_0$, and $B_0^0(x) \equiv 1$ because $\zeta_{-1} \le x < \zeta_0$

$$B_{-1}^{1}(x) = \frac{x-0}{0-0}(0) + \frac{0.5-x}{0.5-0}(1)$$

$$B_{-1}^{1}(x) = \begin{cases} \dfrac{0.5-x}{0.5} & ; 0 \le x < 0.5 \\ 0 & ; \quad else \end{cases} \tag{E.4}$$

B-spline basis function for $j = 1$, $B_0^1(x)$ is

$$B_0^1(x) = \frac{x-\zeta_0}{\zeta_1-\zeta_0}B_{k,0}^{0}(z_k) + \frac{\zeta_2 - x}{\zeta_2-\zeta_1}B_1^0(x)$$

$$B_0^1(x) = \frac{x-0}{0.5-0}(1) + \frac{1-x}{1-0.5}(1)$$

$$B_0^1(x) = \begin{cases} \dfrac{x}{0.5} & ; 0 \le x < 0.5 \\ \dfrac{1-x}{0.5} & ; 0.5 \le x < 1 \end{cases} \tag{E.5}$$

B-spline basis function for $j = 2$, $B_1^1(x)$ is

$$B_1^1(x) = \frac{x-\zeta_1}{\zeta_2-\zeta_1}B_{k,1}^{0}(z_k) + \frac{\zeta_3 - x}{\zeta_3-\zeta_2}B_2^0(x)$$

$$B_1^1(x) = \frac{x-0.5}{1-0.5}(1) + \frac{1-x}{1-1}(0)$$

$$B_1^1(x) = \begin{cases} \dfrac{x-0.5}{0.5} & ; 0.5 \le x < 1 \\ 0; \ else \end{cases} \tag{E.6}$$

$$\boldsymbol{Z} = \boldsymbol{B^1}(x) = \left[\begin{cases} \dfrac{0.5-x}{0.5} & ; 0 \le x < 0.5 \\ 0 & ; \quad else \end{cases} \quad \begin{cases} \dfrac{x}{0.5} & ; 0 \le x < 0.5 \\ \dfrac{1-x}{0.5} & ; 0.5 \le x < 1 \\ 0; else \end{cases} \quad \begin{cases} \dfrac{x-0.5}{0.5} & ; 0.5 \le x < 1 \\ 0; \ else \end{cases} \right]$$

By substituting the value of x, we obtain design matrix of **Z** with dimension 100 × 3

	$B_{-1}^{1}(x)$	$B_0^1(x)$	$B_1^1(x)$
[1,]	1.0000000	0.00000000	0.0000000
[2,]	0.9797980	0.02020202	0.0000000
[3,]	0.9595960	0.04040404	0.0000000

```
[4,]   0.9393939 0.06060606 0.0000000
.....................................
[96,]  0.0000000 0.08080808 0.9191919
[97,]  0.0000000 0.06060606 0.9393939
[98,]  0.0000000 0.04040404 0.9595960
[99,]  0.0000000 0.02020202 0.9797980
[100,] 0.0000000 0.00000000 1.0000000
```

where every column denotes B-spline basis function. It can be drawn as

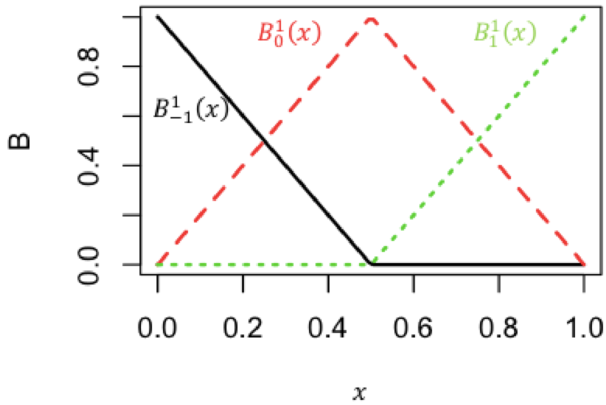

Figure 2. B-spline basis function for $r = 1$ and $p = 1$

The design matrix **Z** is equivalent to design matrix in linear regression. We can fit the curve using least square estimation:

$$\widehat{\boldsymbol{\beta}} = (\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2) = (0.902, 0.035, -0.928)'$$

and we get fitted value of $\widehat{\boldsymbol{y}}$ as $\widehat{\boldsymbol{y}} = \boldsymbol{B}(\boldsymbol{z})\widehat{\boldsymbol{\beta}}$ and the fitted values is presented in fitted curve as:

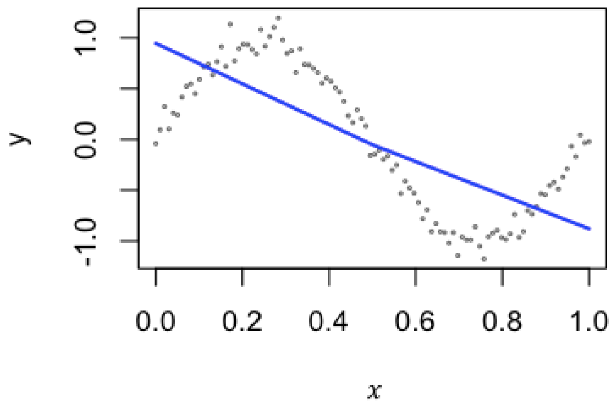

Figure 3. Observed versus fitted values for $r = 1$.

The blue line is the fitted curve based on B-spline with single knot and degree $r = 1$. It seems over smooth. We can improve the fitted line by increasing the degree or the number of knots. Here in this example we increase number of degree becomes $r = 2$.

B-spline with piecewise polynomial with degree $r = 2$ (quadratic)

We compute the design matrix which is equivalent to $\boldsymbol{Z}$

$$\boldsymbol{Z} = \left[\boldsymbol{B}^r_{-r}(x), \boldsymbol{B}^r_{(1-r)}(x), \dots, \boldsymbol{B}^r_p(x)\right] = [\boldsymbol{B}^2_{-2}(x), \boldsymbol{B}^2_{-1}(x), \boldsymbol{B}^2_0(x), \boldsymbol{B}^r_1(x)]$$

with the knot vector $\zeta = (0, 0,0, 0.5, 1, 1, 1)'$.

B-spline basis function for $j = 0$, $B^1_{-2}(x)$ is:

$$B^2_{-2}(x) = \frac{x - \zeta_{-2}}{\zeta_0 - \zeta_{-2}} B^1_{-2}(x) + \frac{\zeta_1 - x}{\zeta_1 - \zeta_{-1}} B^1_{-1}(x) \tag{E.7}$$

where $B^1_{-2}(x) \equiv 0$ because $\zeta_{-2} = \zeta_{-1}$ ($i.e.,$ local support property), and from linear piecewise polynomial (E.4) in previous we get:

$$B^1_{-1}(x) = \begin{cases} \dfrac{0.5 - x}{0.5} & ; 0 \le x < 0.5 \\ 0 & ; \quad else \end{cases}$$

Hence,

$$B^2_{-2}(x) = \frac{x - 0}{0 - 0}(0) + \frac{0.5 - x}{0.5 - 0}\left(\frac{0.5 - x}{0.5}\right) \tag{E.8}$$

$$B^2_{-2}(x) = \begin{cases} \left(\dfrac{0.5 - x}{0.5}\right)^2 & ; 0 \le x < 0.5 \\ 0 & ; \quad else \end{cases}$$

B-spline basis function for $j = 1$, $B^2_{-1}(x)$ is

$$B^2_{-1}(x) = \frac{x - \zeta_{-1}}{\zeta_1 - \zeta_{-1}} B^1_{-1}(x) + \frac{\zeta_2 - x}{\zeta_2 - \zeta_0} B^1_0(x) \tag{E.9}$$

Note from (E.4 and E.5) we get :

$$B^1_{-1}(x) = \begin{cases} \dfrac{0.5 - x}{0.5} & ; 0 \le x < 0.5 \\ 0 & ; \quad else \end{cases}$$

$$B^1_0(x) = \begin{cases} \dfrac{x}{0.5} & ; 0 \le x < 0.5 \\ \dfrac{1 - x}{0.5} & ; 0.5 \le x < 1 \end{cases}$$

Hence:

$$B^2_{-1}(x) = \frac{x - 0}{0.5 - 0}\left(\frac{0.5 - x}{0.5}\right) + \frac{1 - x}{1 - 0}\left(\frac{x}{0.5}\right) \text{ for } 0 \le x < 0.5, \text{and}$$

$$B_{-1}^2(x) = \frac{x-0}{0.5-0}(0) + \frac{1-x}{1-0}\left(\frac{1-x}{0.5}\right) \text{ for } 0.5 \le x < 1$$

$$B_{-1}^2(x) = \begin{cases} \frac{x}{0.5}\left(\frac{0.5-x}{0.5}\right) + (1-x)\left(\frac{x}{0.5}\right) \;; \text{ for } 0 \le z_k < 0.5 \\ (1-x)\left(\frac{1-x}{0.5}\right); \quad \text{for } 0.5 \le z_k < 1 \\ 0; \quad else \end{cases} \tag{E.10}$$

B-spline basis function for $j = 2$, $B_0^2(x)$ is

$$B_0^2(x) = \frac{x-\zeta_0}{\zeta_2-\zeta_0}B_0^1(x) + \frac{\zeta_3-x}{\zeta_3-\zeta_1}B_1^1(x) \tag{E.11}$$

Note that from (E.5 and E.6) we have

$$B_{k,0}^1(z_k) = \begin{cases} \frac{x}{0.5} \;; 0 \le x < 0.5 \\ \frac{1-z_k}{0.5}; 0.5 \le x < 1 \end{cases}$$

$$B_1^1(x) = \begin{cases} 0 \quad ; 0 \le x < 0.5 \\ \frac{x-0.5}{0.5} \;; 0.5 \le x < 1 \\ 0 \quad ;\; else. \end{cases}$$

Hence,

$$B_0^2(x) = \frac{x-0}{1-0}\left(\frac{x}{0.5}\right) + \frac{1-x}{1-0.5}(0) \text{ for } 0 \le x < 0.5$$

$$B_0^2(x) = \frac{x-0}{1-0}\left(\frac{1-x}{0.5}\right) + \frac{1-x}{1-0.5}\left(\frac{x-0.5}{0.5}\right) \text{ for } 0.5 \le x < 1$$

$$B_0^2(x) = \begin{cases} x\left(\frac{x}{0.5}\right) \quad ; 0 \le x < 0.5 \\ x\left(\frac{1-x}{0.5}\right) + \frac{1-x}{0.5}\left(\frac{x-0.5}{0.5}\right); 0.5 \le x < 1 \\ 0 \quad ;\; else \end{cases} \tag{E.12}$$

B-spline basis function for $j = 3$, $B_1^2(x)$ is

$$B_1^2(x) = \frac{x-\zeta_1}{\zeta_3-\zeta_1}B_1^1(x) + \frac{\zeta_4-x}{\zeta_4-\zeta_2}B_2^1(x) \tag{E.13}$$

Note that from $(E.6)$ we have

$$B_{k,1}^1(z_k) = \begin{cases} 0; 0 \le z_k < 0.5 \\ \frac{z_k-0.5}{0.5} \;; 0.5 \le z_k < 1 \\ 0;\; else \end{cases}$$

IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 7 Issue 11, November 2020
ISSN (Online) 2348 – 7968 | Impact Factor (2020) – 6.72
www.ijiset.com

and $B_2^1(x) \equiv 0$ because $\zeta_2 = \zeta_4$ i.e., local support

Hence,

$$B_1^2(x) = \frac{x-0.5}{1-0.5}\left(\frac{x-0.5}{0.5}\right) + \frac{\zeta_4 - x}{\zeta_4 - \zeta_2}(0) \text{ for } 0.5 \le x < 1$$

$$B_1^2(x) = \begin{cases} 0 & ; 0 \le x < 0.5 \\ \left(\frac{x-0.5}{0.5}\right)^2 & ; 0.5 \le x < 1 \\ 0 & ;\ else \end{cases} \qquad \text{(E.14)}$$

$$\boldsymbol{Z_k} = \boldsymbol{B_k^2}(x) = \begin{bmatrix} \begin{cases} \left(\frac{0.5-x}{0.5}\right)^2 & ; 0 \le x < 0.5 \\ 0 & ; otherwise \end{cases} \\ \begin{cases} \frac{x}{0.5}\left(\frac{0.5-x}{0.5}\right) + (1-x)\left(\frac{x}{0.5}\right) & ; 0 \le x < 0.5 \\ (1-x)\left(\frac{1-x}{0.5}\right) & ; 0.5 \le x < 1 \\ 0 & ; otherwise \end{cases} \\ \begin{cases} x\left(\frac{x}{0.5}\right) & ; 0 \le x < 0.5 \\ x\left(\frac{1-x}{0.5}\right) + \frac{1-x}{0.5}\left(\frac{x-0.5}{0.5}\right) & ; 0.5 \le x < 1 \\ 0 & ; otherwise \end{cases} \\ \begin{cases} 0 & ; 0 \le x < 0.5 \\ \left(\frac{x-0.5}{0.5}\right)^2 & ; 0.5 \le x < 1 \\ 0 & ; otherwise \end{cases} \end{bmatrix}^T$$

By substituting the value of x, we obtain design matrix of **Z** with dimension 100 × 4

	$B_{-2}^2(x)$	$B_{-1}^2(x)$	$B_0^2(x)$	$B_1^2(x)$
[1,]	1.0000000	0.0000000000	0.0000000000	0.0000000
[2,]	0.9600041	0.0397918580	0.0002040608	0.0000000
[3,]	0.9208244	0.0783593511	0.0008162432	0.0000000
[4,]	0.8824610	0.1157024793	0.0018365473	0.0000000
....				
[96,]	0.0000000	0.0032649730	0.1518212427	0.8449138
[97,]	0.0000000	0.0018365473	0.1157024793	0.8824610
[98,]	0.0000000	0.0008162432	0.0783593511	0.9208244
[99,]	0.0000000	0.0002040608	0.0397918580	0.9600041
[100,]	0.0000000	0.0000000000	0.0000000000	1.0000000

IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 7 Issue 11, November 2020
ISSN (Online) 2348 – 7968 | Impact Factor (2020) – 6.72
www.ijiset.com

where every column denotes B-spline basis function. It can be drawn as:

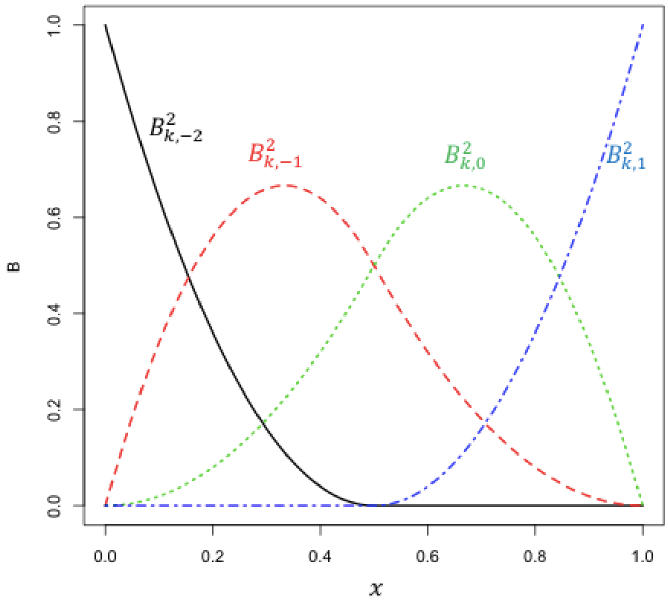

Figure 4. B-spline basis function for $r = 2$ and $p = 1$

The design matrix **Z** is equivalent to design matrix in linear regression. We can fit the curve using least square estimation and we get the control points:

$$\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3) = (-0.08356, 2.00255, -1.98404, 0.08566)'$$

and we get fitted value of $\hat{\boldsymbol{y}}$ as $\hat{\boldsymbol{y}} = \boldsymbol{B}(\boldsymbol{z})\hat{\boldsymbol{\beta}}$ and the fitted values is presented in fitted curve as:

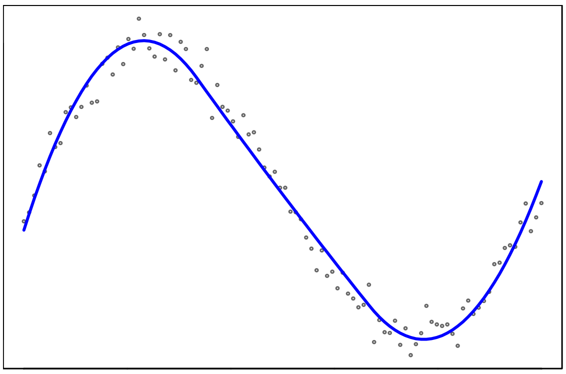

Figure 5. Observed versus fitted values for $r = 2$.

The blue line is the fitted curve based on B-spline with single knot and degree=2.

We present three examples below:

(a) Linear

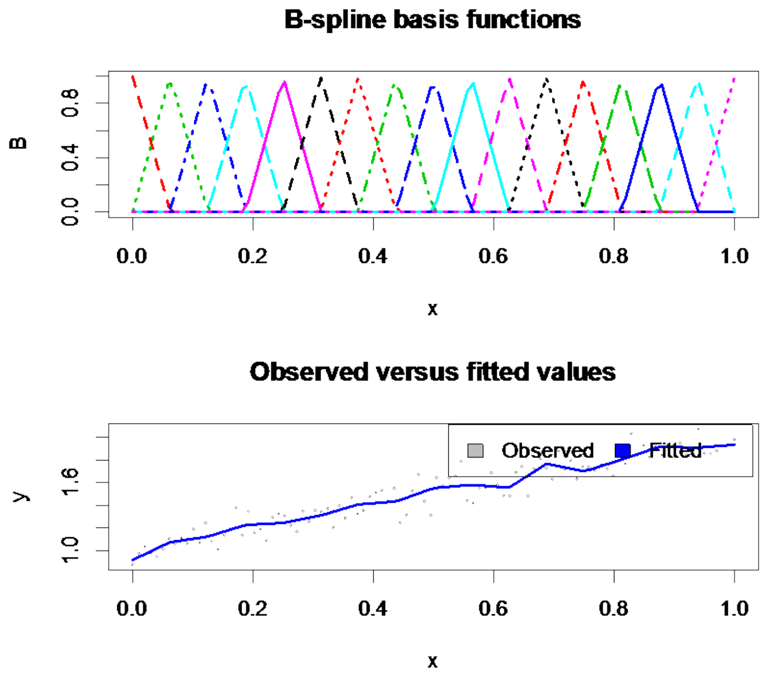

Figure 6. Approximate the linear model

(b) Simple nonlinear

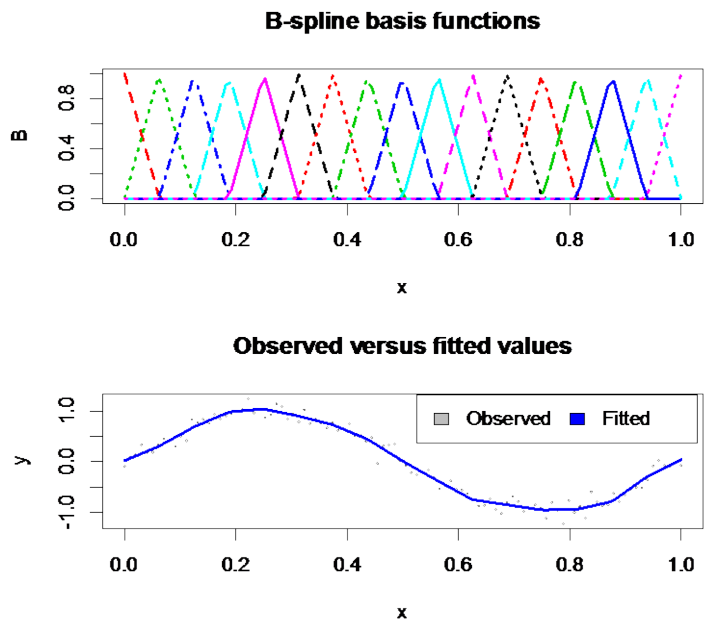

Figure 7. Approximate the simple nonlinear

(c) Seasonal

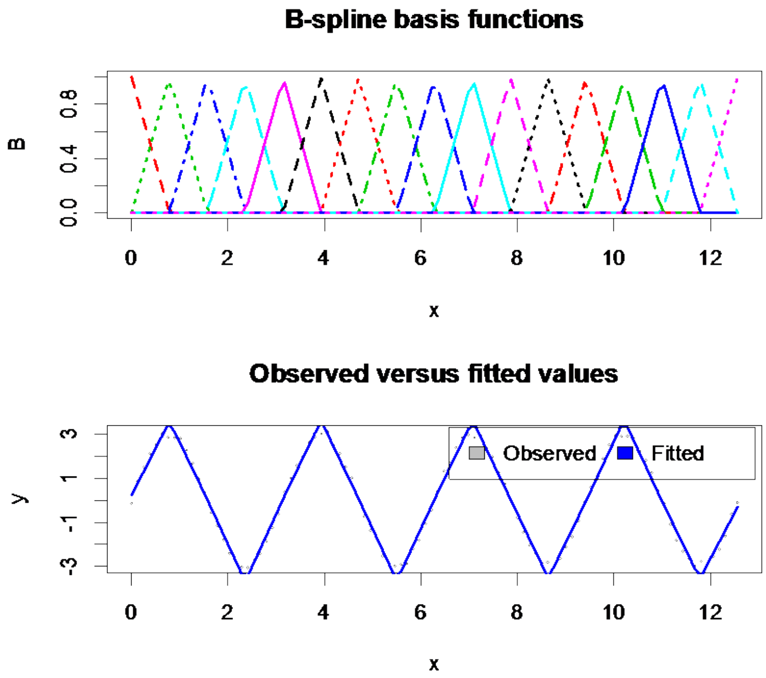

Figure 8. Approximate the seasonal pattern

Given the simple examples we found our method provides a good result especially for nonlinear regression. The data observation in Figures (6-8) are approximated well by our approach. It is quite simple and provides a good estimate of nonlinear curve. This approach does not need much time to computation process.

Conclusion

B-spline is a powerful regression technique, particularly for nonlinear regression. It works by combining several basis functions that have local support. However, defining the number of knots and their positions. To deal with the position of the knots, they are commonly placed uniformly with similar space. However, the number of knots may need more attention. Here we propose to use Sturges rule to create optimal number of bins, and the bins boundary represents knots. We apply our approach to three different data set linear, simple nonlinear, and seasonal patterns. The results are quite good for a simple approximation with the minimum computational time required.

References

[1] F. Provost and T. Fawcett, "Data Science and its Relationship to Big Data and Data-Driven Decision Making," *Big Data,* pp. 1-9, 2013.

[2] S. Zillner, T. Becker, R. Munné, K. Hussain, S. Rusitschka, H. Lippell, E. Curry and A. Ojo, "Big Data-Driven Innovation in Industrial Sectors," in *A Roadmap for Usage and Exploitation of Big Data in Europe New Horizons for a Data-Driven Economy*, Madrid, Springer Open, 2016, pp. 169-178.

[3] F. Rousseaux, "BIG DATA and Data-Driven Intelligent Predictive Algorithms to support creativity in Industrial Engineering," *Computers & Industrial Engineering,* vol. 112, pp. 459-465, 2017.

[4] D. Aydin, "A Comparison of the Nonparametric Regression Models using Smoothing Spline and Kernel Regression," *World Academy of Science, Engineering and Technology* , vol. 36, pp. 1-5, 2007.

[5] J. Racine, "A primer on Regression Splines," 1 11 2019. [Online]. Available: https://cran.r-project.org/web/packages/crs/vignettes/spline_primer.pdf. [Accessed 2020 11 1].

[6] A. Perperoglou, W. Sauerbrei, M. Abrahamowicz and M. Schmid, "A review of spline function procedures in R," *BMC Medical Research Methodology,* pp. 19-46, 2019.

[7] McGraw-Hill, Dictionary of Scientific & Technical Terms, 6E, The McGraw-Hill Companies, 2003.

Appendix

```
Bspline<-function(x,y,d,c){

R<-max(x)-min(x)
K<-1+3.33*log(n)
K1<-round(K,0)
I<-R/K1
```

```
Knot<-rep(0,K1)
for(i in 1:K1){
Knot[i]<-M1+(i-1)*I
}

basis <- function(x, degree, i, knots) {
  if(degree == 0){
    B <- ifelse((x >= knots[i]) & (x < knots[i+1]), 1, 0)
  } else {
    if((knots[degree+i] - knots[i]) == 0) {
      alpha1 <- 0
    } else {
      alpha1 <- (x - knots[i])/(knots[degree+i] - knots[i])
    }
    if((knots[i+degree+1] - knots[i+1]) == 0) {
      alpha2 <- 0
    } else {
      alpha2 <- (knots[i+degree+1] - x)/(knots[i+degree+1] - knots[i+1])
    }
    B <- alpha1*basis(x, (degree-1), i, knots) + alpha2*basis(x, (degree-
 1), (i+1), knots)
  }
return(B) }

bs <- function(x, degree, interior.knots=Knot, intercept=FALSE,
 Boundary.knots = c(min(x),max(x))) {
  if(missing(x)) stop("You must provide x")
  if(degree < 1) stop("The spline degree must be at least 1")
  Boundary.knots <- sort(Boundary.knots)
  interior.knots.sorted <- NULL
  if(!is.null(interior.knots)) interior.knots.sorted <-
 sort(interior.knots)
  knots <- c(rep(Boundary.knots[1], (degree+1)), interior.knots.sorted,
 rep(Boundary.knots[2], (degree+1)))
  K <- length(interior.knots) + degree + 1
  B.mat <- matrix(0,length(x),K)
  for(j in 1:K) B.mat[,j] <- basis(x, degree, j, knots)
  if(any(x == Boundary.knots[2])) B.mat[x == Boundary.knots[2], K] <- 1
  if(intercept == FALSE) {
    return(B.mat[,-1])
  } else {
    return(B.mat)
  }
}

B <- bs(x, degree=d, intercept = TRUE,  interior.knots=Knot,
 Boundary.knots=c(min(x), max(x)))
par(omi = c(.5, .5, .5, .5),           # Set outer margin at .5 inches
  mai = c(.8, .8, .8, .8),             # Set inner margin btwn figure &
 plot
  fin = c(6, 4))
par(mfrow = c(2,1))
matplot(x, B, type="l", lwd=2,  xlab="x", main="B-spline basis functions")
model <- lm(y~B-1)
yhat<-B%*%solve(t(B)%*%B+c*diag(ncol(B)))%*%t(B)%*%y
plot(x, y, cex=.25, col="grey40",  xlab="x", main="Observed versus fitted
 values")
lines(x, yhat, lwd=2, col="blue")
 legend("topright", inset=.01,
   c("Observed","Fitted"), fill=c("gray","blue"), horiz=TRUE)
```

```
}

Note:
d: degree
c: smoothing parameter
```