A Study On Spline Regression

Arman Terzyan

December 2019

Abstract

We examine the use of spline regression and its various forms of implementation. We seek to improve results achieved from both linear and polynomial regression by exploiting the beneficial properties of a spline. A full synthetic example will be shown in R along with a description of the advantages of using this nonparametric method.

1 Introduction

The goal in a regression task is to estimate the relationship between a dependent variable and one or more independent variables. The most popular parametric approaches are linear and polynomial regression. These methods have benefits ranging from simplicity to interpretability, though issues arise from forcing a global structure on a data set. Spline regression offers a stable yet flexible alternative where a model is learned by merging several individual models from segmented portions of the data set.

2 Parametric Approaches

Parametric methods make several assumptions about the underlying data set prior to modeling. Specifically for regression, a parametric method has a finite predetermined number of parameters that must be approximated. When this assumption is ultimately correct, these methods will generally lead to higher statistical power compared to their nonparametric counterparts. However, the lack of flexibility can lead to poor models.

2.1 Linear Regression

Linear regression is the process of estimating the relationship between a set of predictors and a response with a linear function. The number of parameters to estimate depends on the number of predictors used in the model. Hence in the simplest case of one predictor, we will estimate two parameters. These parameters can be thought of as the intercept term and the slope for the predictor showing how the response varies as the value of that predictor varies.

$$Y = \beta_0 + \beta_1 X + \epsilon$$

The parameters β_0, β_1 are found by minimizing the sum of squared residuals $\displaystyle\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$. This can be done by solving the normal equation $(X^T X)^{-1} X^T Y$ or through an iterative process like gradient descent. Given that it's uncommon to see a linear relationship in practice, this method can lead to high bias from underfitting.

2.2 Polynomial Regression

Polynomial regression adds a layer of complexity by estimating the relationship between a set of predictors and a response with an n^{th} degree polynomial. Despite the added complexity, polynomial regression is just a special case of multiple linear regression. The idea follows by sequentially raising a predictor to a power until reaching the chosen degree n and then incorporating these power terms into the model as new predictors. The formula would be

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + ... + \beta_n X^n + \epsilon$$

Note that the function is still linear in the unknown parameters. Though polynomials can fit a much wider range of functions, there is a high potential for overfitting.

3 Spline

Spline regression does not have a set amount of parameters to estimate. The model will be learned from the data. Despite generally requiring more data relative to a parametric approach, splines offer an alternative that can give a good balance between bias and variance. It is also more applicable to real life situations which typically have localized patterns in addition to outliers.

3.1 Basics

A spline can be thought of as two or more curves that are joined together at some predetermined points. In essence, it is a function that is built piecewise by polynomials. Ideally the polynomials will be of a low degree.

3.2 History

Splines were not originally developed for use in mathematics. They were however vastly utilized in the design of ships. Builders would place weights along some path and bend a rod through those points allowing for a smooth curve. It has also been applied in airplane design and trajectory plotting. The use in mathematics was not seen until the 1950s.

3.3 Classical Construction

The process of a spline regression will be described step by step below.

1. Divide the data set into a chosen number of segments. The point(s) at which the data is split will be referred to as a knot. Knots will be denoted by ξ_i.

2. Model each section between each pair of knots with a polynomial of a chosen degree, d. If $d = 1$ then the spline will be made with linear functions. The most common case is $d = 3$ which is referred to as a cubic spline.

3. To achieve smoothness, continuity up to the $d - 1$ derivative is desired along with the piecewise functions being equal at the knots up to the $d - 1$ derivative.

4. To force the curves to join at the knots, the translated rectified linear unit function will be employed.

$$(X - \xi_i)_+^d = \begin{cases} 0 & X < \xi_i \\ (X - \xi_i)^d & X \geq \xi_i \end{cases}$$

5. A basis is needed in order to actually fit the model. Several bases are available but a classical approach is the truncated power basis. Our spline function can be expressed as

$$Y = \sum_{i=1}^{d+k+1} \beta_i h_i(x)$$

with basis $h_1(x) = 1, h_2(x) = x, \cdot \cdot \cdot, h_{d+1}(x) = x^d, h_{d+1+k}(x) = (x - \xi_k)_+^d$. k represents the number of knots.

6. Continuing with the truncated power basis, we will build the design matrix. Suppose we wanted a cubic spline with two knots placed in a data set with n observations. The design matrix would be

$$\begin{pmatrix} 1 & x_1 & x_1^2 & x_1^3 & (x_1 - \xi_1)_+^3 & (x_1 - \xi_2)_+^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 & x_n^3 & (x_n - \xi_1)_+^3 & (x_n - \xi_2)_+^3 \end{pmatrix}$$

As we have a cubic spline with two knots, there will be six parameters to estimate with the intercept term.

$$\begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \beta_5 \\ \beta_6 \end{pmatrix}$$

The coefficients can be estimated with linear regression.

3.3.1 Knot Selection

The placement of knots as well as the number of knots chosen is crucial to the quality of the spline model. Too few or too many knots can lead to underfitting or overfitting of the model respectively. In addition, placing knots at appropriate points can greatly improve results. The simplest method is trial and error. Trying a knot at each quantile is a good start. Going further, cross validation can be utilized to see what number of knots minimized some chosen error function. Cross validation follows by separating the data set into what are called folds. Choose one fold to be the validation set while the remaining will be the training set. With some chosen number of knots, build your model on the training set and calculate the error on the validation set. Keep repeating this process until each fold has had an opportunity to be the validation set. After completing the repetitions, calculate the average error for that chosen number of knots and repeat the whole process with several different amounts of knots. Choose the number that minimizes error. The most sophisticated method, which can allow for avoiding any specific knot selection, is to use smoothing splines. With smoothing splines a knot is placed at each data point. Least squares minimization is performed but an additional penalty term is added to control for overfitting. It is similar to a regularized regression technique. For a cubic spline, we seek to minimize

$$\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda \int_a^b (f''(x))^2 dx$$

for some chosen value of λ. Smaller values of λ lead to bias while high values of λ lead to high variance.

3.4 Variants

A common issue with both spline and polynomial regression is the erratic behavior that occurs at the boundaries. In the case of interpolation, this behavior is known as Runge's phenomenon. To mitigate this problem, we implement natural splines. A natural spline adds constraints in the boundary regions forcing the polynomial in that segment to be of lower degree relative to the selected degree for the regression spline. Specifically the altered polynomial will be of degree $\frac{d-1}{2}$. Hence if a cubic spline model was to be built, the functions in the boundaries would be linear thereby making the second derivative equal to zero. Thus the quadratic and cubic term for observations in those regions would also be set equal to zero in the process of building a design matrix.

In the previous section, we constructed a regression spline with the classical truncated power basis expansion. Computationally this approach could result in slow run times along with matrix errors. For this reason, most regression software utilize a B-spline basis which are far more efficient.

4 Code Implementation

We now show an implementation of spline regression with a toy data set. In the following section we will visually compare the resulting fit to that of linear and polynomial regression.

```
##Spline regression##

set.seed(123)#set seed
x=runif(100,0,10)#build toy dataset and plot
y=sin(x)+ .25*rnorm(100)
plot(x,y)
knots=c(2,5,8)#choose knots through visual inspection
abline(v=knots,col=”darkgreen”)#plot knots

df=data.frame(cbind(x,y))#dataframe of toy dataset values
colnames(df)=c('x','y')

degree=3#set degree
degreeX=outer(df$x,1:degree,”^”)#build design matrix of data to esimate Beta values
knotsX=outer(df$x,knots,”>”)* #check if value of x is in interval
   outer(df$x,knots,”-”)^degree
ones=rep(1,nrow(df))
X=cbind(ones,degreeX,knotsX)
mydf=data.frame(cbind(X,y))#combine with response
```

```
betas=solve(t(X)%*%X)%*%(t(X)%*%y)#solve for coefficients with normal equation

#build function to create new data points that will be used for plotting the spline
newPoints=function(k,points,degree){
  degreeX=outer(points,1:degree,"^")
  knotsX=outer(points,k,">")*
    outer(points,k,"-")^degree
  ones=rep(1,length(points))
  X=cbind(ones,degreeX,knotsX)
  X=as.data.frame(X)
  return(X)
}

x.new=seq(-1,11,by = .01)#new data points
points(x.new,as.matrix(newPoints(knots,x.new,degree))%*%b          etas,col="darkgreen",t
```

5 Results

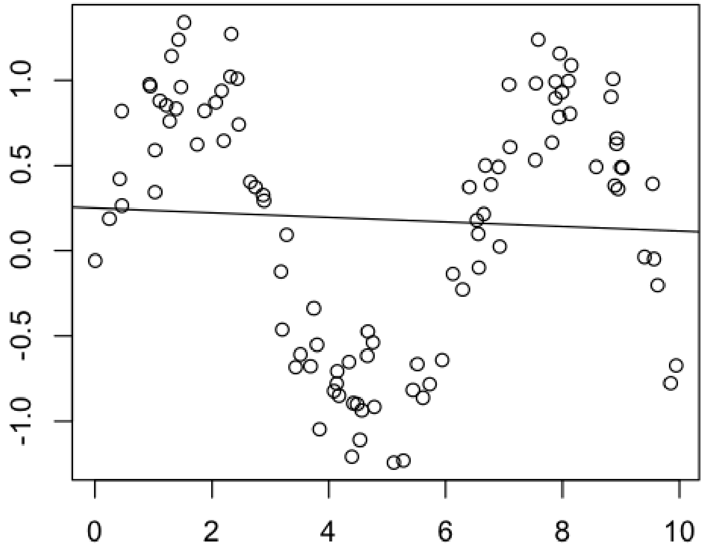

Figure 1: Linear regression

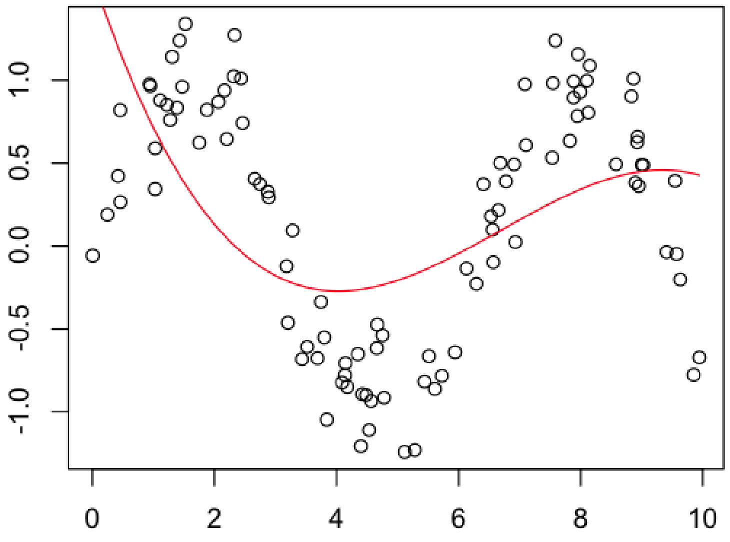

Figure 2: Cubic polynomial regression

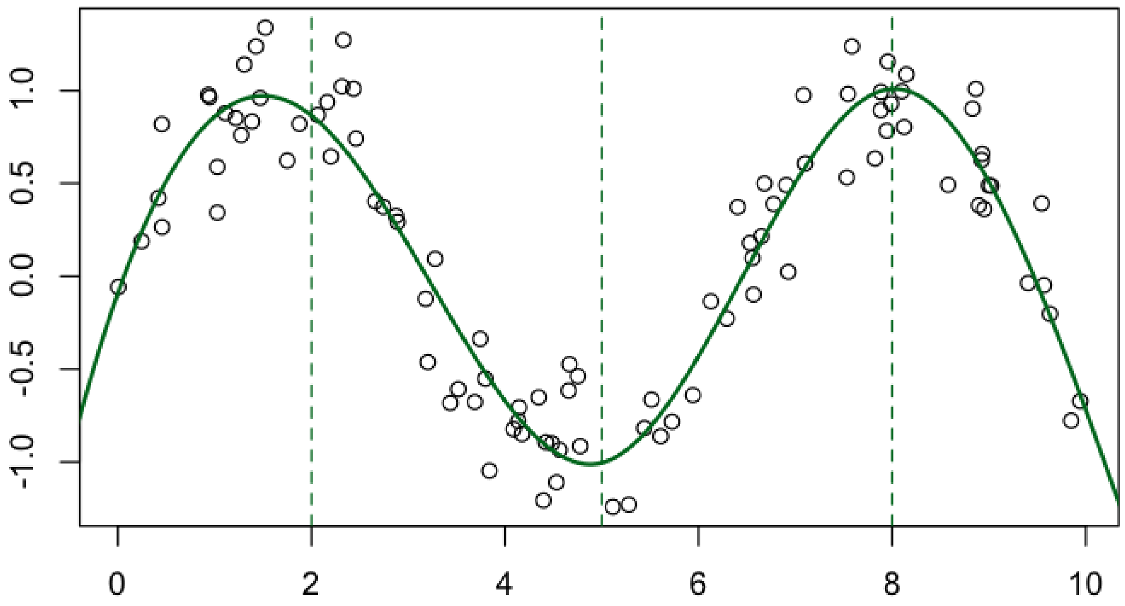

Figure 3: Cubic spline regression

6 Conclusion

In this work, we studied the usage of spline regression. Though there exist some caveats when working with a nonparemtric method, spline regression outperforms linear and polynomial regression in most situations. The ability to localize a dataset brings about flexibility in addition to a good balance between bias and variance.

7 References

1. Hastie, T., Tibshirani, R.,, Friedman, J. (2001). The Elements of Statistical Learning. New York, NY, USA: Springer New York Inc..

2. Wasserman, Larry. All of Nonparametric Statistics. Springer, 2010.

3. A History of the Spline. (2014, September 16). Retrieved from https://www.izenda.com/from-ships-to-data-a-history-of-the-spline/.