Wrocław University of Science and Technology

Splines in Statistical Learning

by

Segun Light Jegede *and* Isaac Akoji Paul

from:
Department of Pure and Applied Mathematics
Data Mining Course

2nd of November, 2020

HR EXCELLENCE IN RESEARCH

What are Splines?

- What are Splines?
- Piecewise Polynomials and Splines
- Spline of Degree k
- Natural Cubic Spline
- Smoothing Spline
- Degree of Freedom and Smoother Matrices
- Choosing Spline Parameters
- Integrated Squared Predicted Error (EPE)
- Multidimensional Splines
- B-Splines
- Case Study
- Conclusion

References

R Code for Data Illustration

R Code for Case Study

Wroclaw University of Science and Technology

What are Splines?

- a curve
- the first mathematical reference to spline is in Schoenberg's book chapter titled "Contributions to the problem of approximation of equidistant" - 1946,
- function specially defined piecewise by polynomials,
- the ideas have their roots in the aircraft and ship-building,
- but have its usefulness across fields such as pharmacokinetics, astrophysics, geophysics and so on.
- applicable in theoretical and applied statistical research
- especially when data requires interpolation and smoothing,

What are Splines? – The History:

- “splines” - the word was originally an East Anglian dialect word.
- the problem – naval architects needed a way to draw a smooth curve through a set of points
- the Solution – place metal weights (called knots; “ducks” by Forrest or “dogs”/“rats” by Schoenberg) at the control points and pass thin wooden strips (called “splines”) through the weights.
- this technique is borrowed from ship-hull design and was used during World War II by British aircraft industry.

What are Splines? – An Illustration

Figure 1: Splines passing through a set pf points called “ducks” or ”knots”

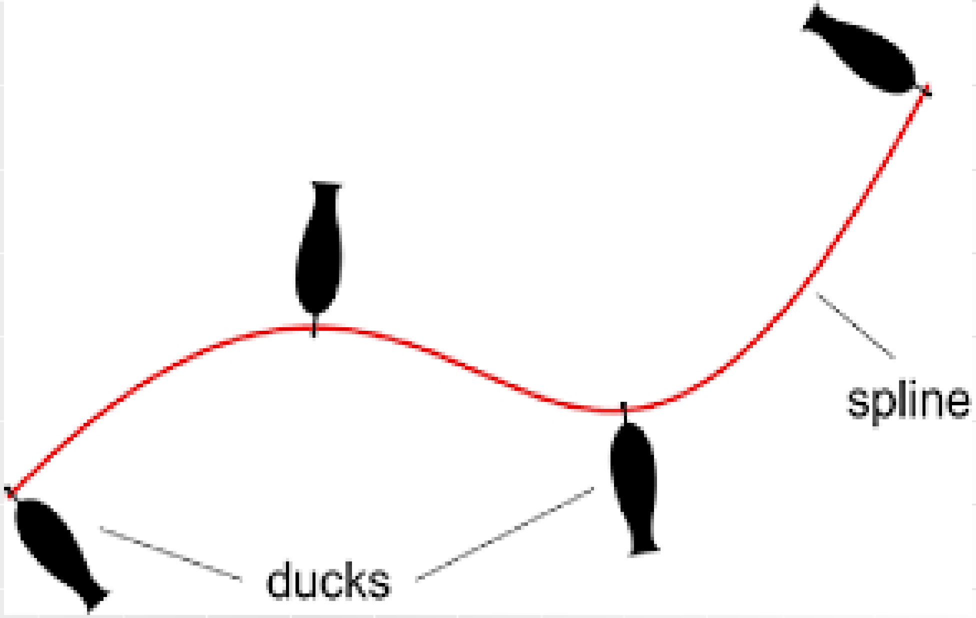
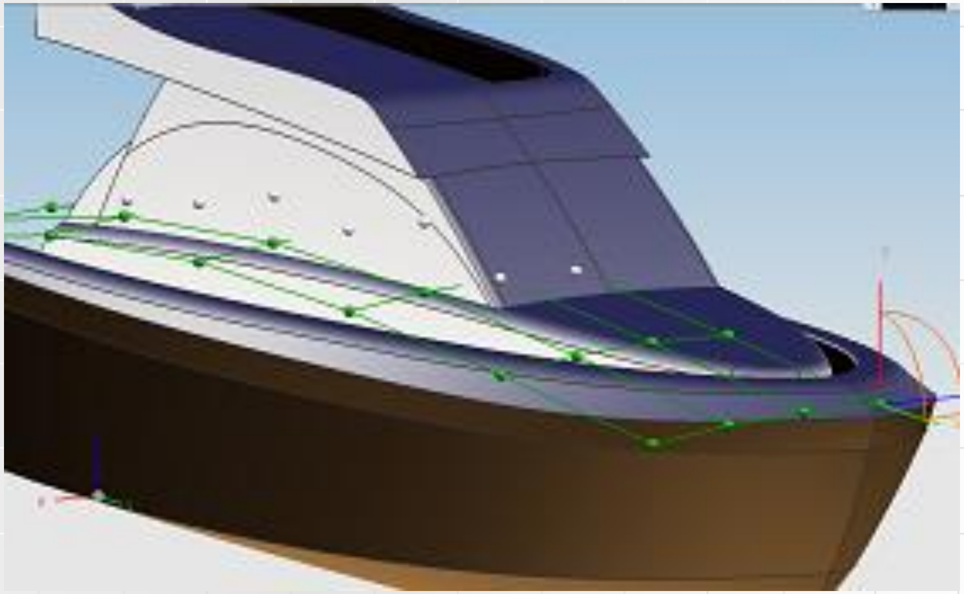


Figure 2: Splines on a Motor Yacht



What are Splines? – The Idea:

- Consider a vector of inputs, X , the independent variable of a model
- Denote $h_m(X): \mathbb{R}^p \rightarrow \mathbb{R}$ the mth transformation of X, with m=1,…,M.

  Then, $f(X)$ is modelled as

$$f(X) = \sum_{m=1}^{M} \beta_m h_m(X) \ldots \ldots \ldots \ldots (i)$$

- with $f(X)$, a linear basis expansion in X
- in short, the idea is to compute transformations of X, and then use linear models in this new models in this new space of derived input features.

What are Splines? – Data Illustration:

❖ consider a model $Y = aX + c$

Where X is age, Y is salary and a is the slope.

❖ polynomial (quadratic) function of the model: $Y = aX^2 + bX + c$

❖ The linear doesn't fit well; the apex of the data is also different from what the quadratic predicts.

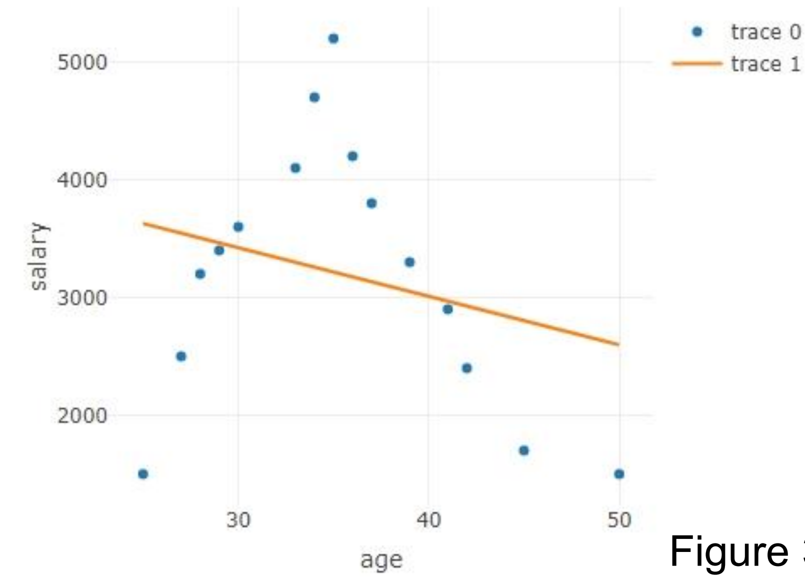


Figure 3

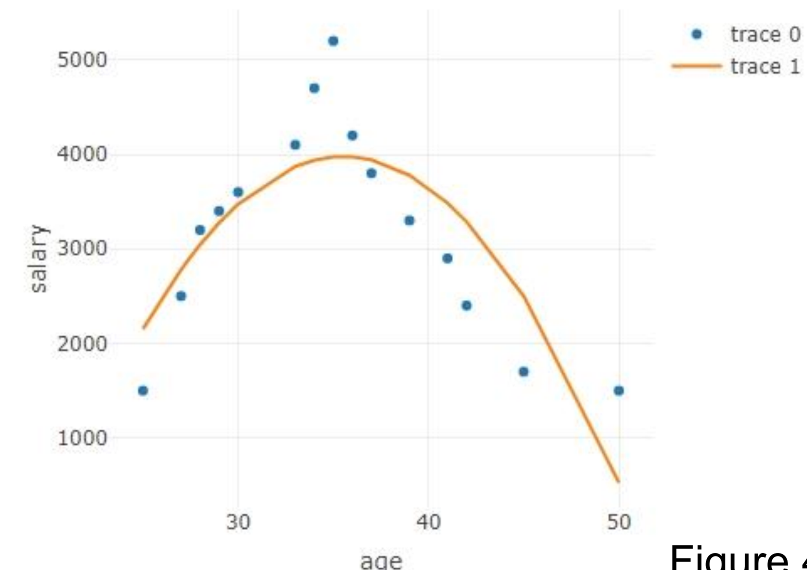


Figure 4

What are Splines? – Data Illustration:

- the real apex is observed around age=35
- place a knot at age=35.
- new function:

$$Y = aX^2 + b(X - 35) \times Xk + c$$

$$X_k = \begin{cases} 0 & X \leq 35 \\ 1 & X > 35 \end{cases}$$

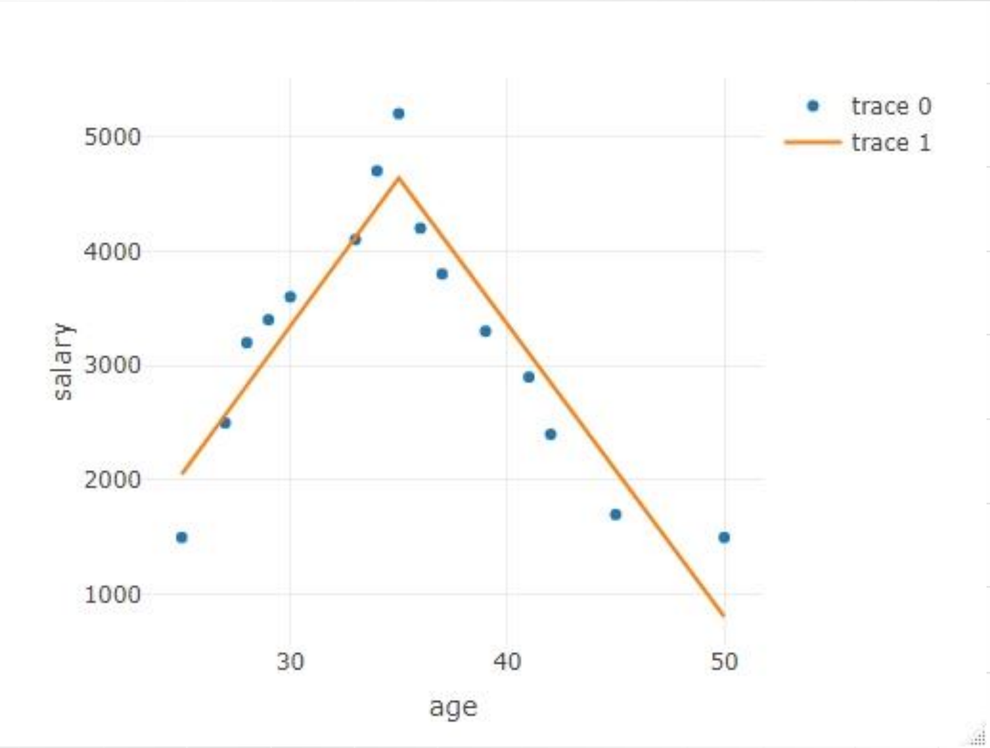


Figure 5: Splines Illustration

Piecewise Polynomials and Splines

- a piecewise polynomial function $f(X)$ is obtained by dividing the domain of f into contiguous interval, and presenting f by a separate polynomial in each interval.
- consider Figure 3 and label each gapped line (knot) as ξ_1 and ξ_2 respectively
- the contiguous interval:

$$h_1(X) = I(X < \xi_1), h_2(X) = I(\xi_1 \leq X < \xi_2), h_1(X) = I(\xi_2 \leq X) \dots \dots \dots \dots (ii)$$

- in each interval, the degree of the fitted line determines the order of the piecewise-polynomial.
- an order-M spline with spline with knot $\xi_j, j = 1,2, \dots, k$, is a piecewise-polynomial of order M, and has continuous derivatives up to order M-2.
- for instance, the piecewise-constant function is an order-1 spline, while the continuous piecewise linear function is an order-2 spline.

Piecewise Polynomials and Splines

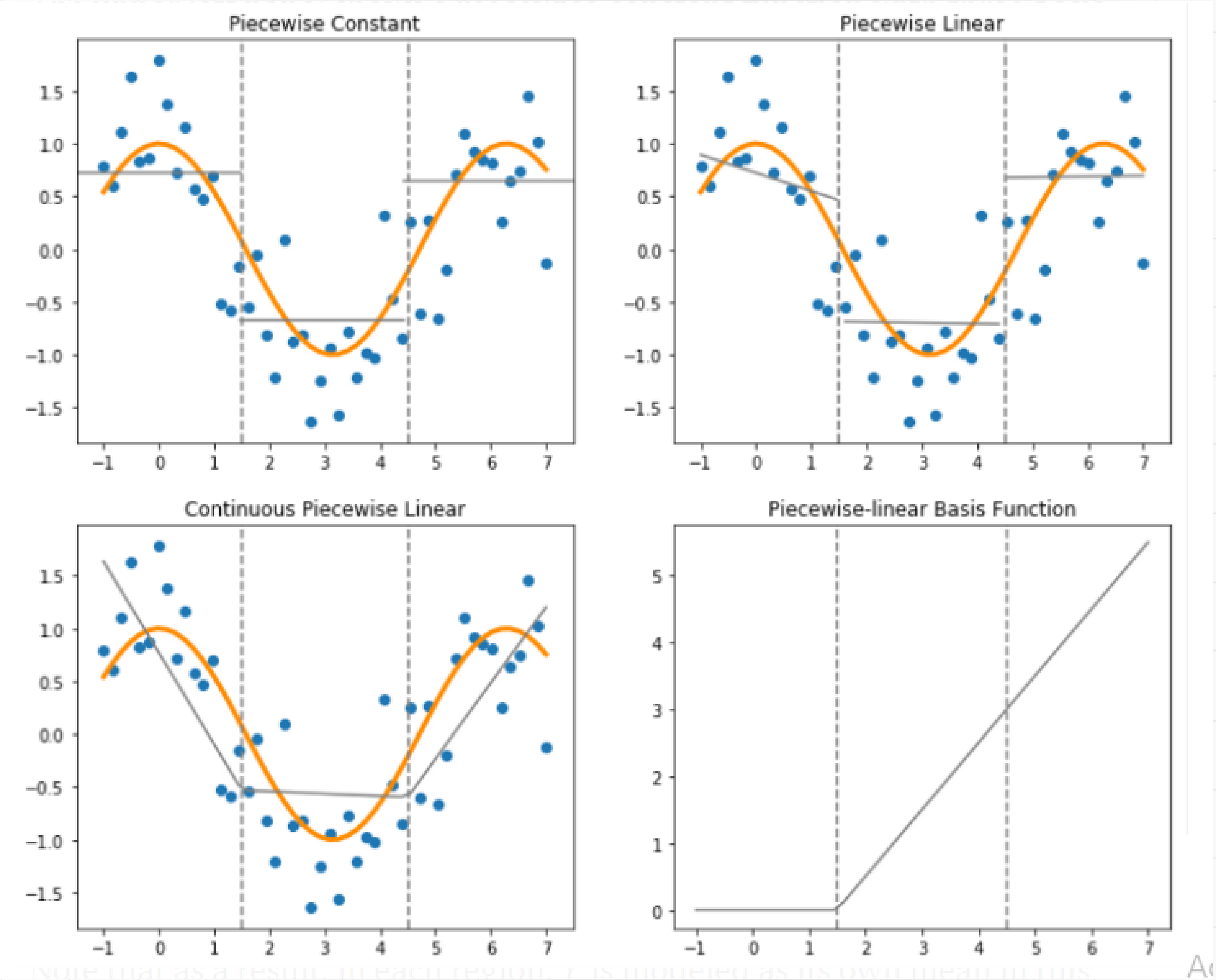


Figure 6: Piecewise Polynomials and Splines

Piecewise Polynomials and Splines

❖ Let's take a case study from a "Polygonal Function"

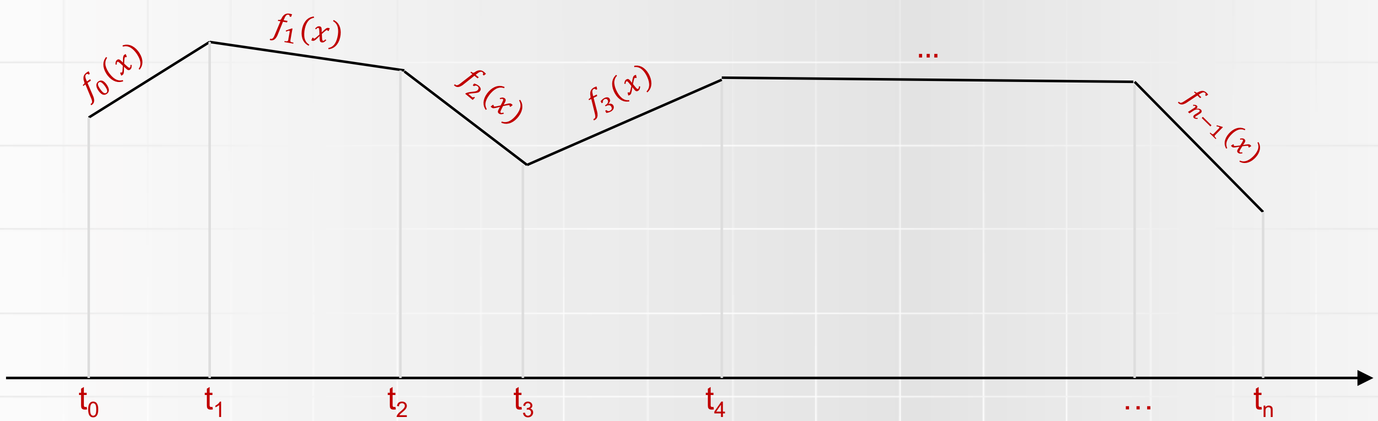


Figure 4: Graph of polygonal function

$$f(x) = \begin{cases} f_0(x) & x \in [t_0, t_1] \\ f_1(x) & x \in [t_1, t_2] \\ \vdots & \\ f_{n-1}(x) & x \in [t_{n-1}, t_n] \end{cases}$$

with $f_i(x) = a_i x + b_i$ ... ... (iii)

❖ such function, $f(x)$, is called piecewise linear.

❖ the polygonal function is a spline of degree 1 which consists of linear polynomials joined together to achieve continuity.

Spline of Degree k

- A function, $f(x)$ is called a spline of degree 1 if:
  - The domain of f is an interval [a, b]
  - $f', f'', \dots, f^{(k-1)}$in each interval.
  - There are points t_i(the knots of S) such that $a = t_0 < t_i < \cdots < t_n = b$ and such that S is a polynomial of degree at most k on each subinterval $[t_i, t_{i+1}]$.
- This leads to different kinds of Spline:
  - Linear Spline: Spline of degree 1
  - Quadratic Spline: Spline of degree 2
  - Cubic Spline: Spline of degree 3
- In practice, the most widely used are up to degree 3
- the fixed knot splines are also known as regression splines

Natural Cubic Spline

- a piece-wise cubic polynomial – a special case of cubic splines
- twice continuously differentiable
- has less tendency to oscillate between data points during interpolation
- a natural cubic spline with k knots is represented by k basis function.
- it adds additional constraints,

$$f_0''(x_0) = 0 \text{ and } f_{n-1}''(x_n) = 0 \ldots \ldots \ldots \ldots (iv)$$

- which assumes linearity outside of boundary knots

Smoothing Spline

- controls complexity of fit by regularization
- generally, spline functions does not exhibit oscillatory behaviour compare to polynomials during interpolation.
- Consider,

$$RSS(f,\lambda) = \sum_{i=1}^{n} \{y_i - f(x_i)\}^2 + \lambda \int \{f''(t)\}^2 dt \ldots \ldots \ldots (v)$$

- search for a function that minimizes penalized residual sum of squares depending on the choice of smoothing parameter, λ

Smoothing Spline

- furthermore, consider n-dimensional set of basis functions for representing the family of natural splines - $N_j(x)$ with $j = 1,2, \dots, n$
- $f(x)$ can be written as $f(x) = \sum_{j=1}^{n} N_j(x)\,\theta_j \dots \dots \dots \dots (vi)$
- Thus, $RSS(\theta, \lambda) = (y - N\theta)^T(y - N\theta) + \lambda\theta^T\Omega_n\theta \dots \dots \dots \dots (vii)$

with $\{N\}_j = N_j(x_i)$ and $\{\Omega_n\}_{jk} = \int N_j''(t) N_k''(t)\,dt$

the solution is a form of the generalized ridge regression presented as:

$$\hat{\theta} = (N^T N + \lambda\Omega_n)^{-1} N^T y \dots \dots \dots \dots (viii)$$

The estimated fitted smoothing spline is given as:

$$\hat{f}(x) = \sum_{j=1}^{n} N_j(x)\,\hat{\theta}_j \dots \dots \dots \dots (ix)$$

Degree of Freedom and Smoother Matrices

- Let $\hat{f}$ be the n-vector of fitted values $\hat{f}(x_i)$ at the training predictors, x_i.

  Then, $\hat{f} = N\hat{\theta} = N\big((N^TN + \lambda\Omega_n)^{-1}N^Ty\big) = S_\lambda y \ldots\ldots\ldots\ldots (x)$

- S_λ, the finite linear operator is known as the smoother matrix

- the effective degree of freedom,

$$df_\lambda = trace(S_\lambda) \ldots\ldots\ldots\ldots (xi)$$

- while λ ranging from 0 to ∞, controls smoothness and restricts effective degrees of freedom, df_λ

- df_λ is strongly connected to the number of knots and depends on the choice of λ.

Choosing Spline Parameters

Choosing the number and locations of the knot:

- ❖ place more knots at points where the function might vary most rapidly
- ❖ in practice, knots are placed in a uniform fashion
- ❖ this implies specifying the desired degrees of freedom and allow the software automatically place the knots at uniform quantiles of the data

Choosing df_λ :

- ❖ create model with different values of df_λ and choose the best fit
- ❖ use of cross validation, CART, MARS or boosting.

Integrated Squared Predicted Error (EPE)

- important in controlling model fit
- the term "variance" accounts for model over-fitting
- the term "bias" accounts for model under-fitting
- EPE controls both and allows us to find the best fit

$$Cov(\hat{f}) = S_\lambda Cov(y) {S_\lambda}^T = S_\lambda {S_\lambda}^T \dots\dots\dots\dots (xii)$$

$$Bias(\hat{f}) = f - E(\hat{f}) = f - S_\lambda f \dots\dots\dots\dots (xiii)$$

$$EPE(\hat{f}_\lambda) = E(y_i - f(x_i))^2 = \sigma^2 + MSE(\hat{f}_\lambda) \dots\dots\dots\dots (xiv)$$

where $MSE(\hat{f}_\lambda) = E[Bias^2\left(\hat{f}_\lambda(x)\right) + \mathrm{Var}\left(\hat{f}_\lambda(x)\right)]$

- creates a trade off between bias and variance
- an approximately unbiased estimator is leave-one-out cross validation curve expressed mathematically as:

$$CV(\hat{f}_\lambda) = \frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{f}_\lambda^{-i}(x_i))^2 \dots\dots\dots\dots (xv)$$

Multidimensional Splines

- they can be seen as generalisation of 1D case
- every one-dimensional approach has it's multi-dimensional counterpart

Consider $x \in \mathbb{R}^2$, the $M_1 \times M_2$ dimensional tensor product basis defined by

$$g_{jk}(x) = h_{1j}(x_1)h_{2k}(x_2), \quad j = 1, \dots, M_1;\ k = 1, \dots, M_2 \ \dots \dots \dots (xvi)$$

can be used for representing a two-dimensional function:

$$g(x) = \sum_{j=1}^{M_1} \sum_{k=1}^{M_2} \theta_{jk} g_{jk}(x) \dots \dots \dots (xvii)$$

where $h_{1j}(x_1)$ and $h_{2k}(x_2)$ are basis function for representing function of

coordinate x_1 and x_2.

Multidimensional Splines

Suppose we have pairs y_i, x_i with $x_i \in \mathbb{R}^d$, and we seek a d-dimensional regression function f(x).

The idea is to set up

$$\min_f \sum_{i=1}^{n} \{y_i - f(x_i)\}^2 + \lambda J[f] \dots\dots\dots (xviii)$$

where J is a natural generalisation of penalty function in 1D case, that is:

$$J[f] = \int\int \left(\left(\frac{\partial^2 f(x)}{\partial x_1^2} \right)^2 + 2\left(\frac{\partial^2 f(x)}{\partial x_1 \partial x_2} \right)^2 + \left(\frac{\partial^2 f(x)}{\partial x_2^2} \right)^2 \right) dx_1 dx_1 \dots\dots\dots (xix)$$

having $x = [x_1, x_2]$.

The solution has the form:

$$f(x) = \beta_0 + \beta^T x + \sum_{j=1}^{n} \alpha_j h_j(x) \dots\dots\dots (xx)$$

where $h_j(x) = \|x - x_j\|^2 log\|x - x_j\|$

Multidimensional Splines

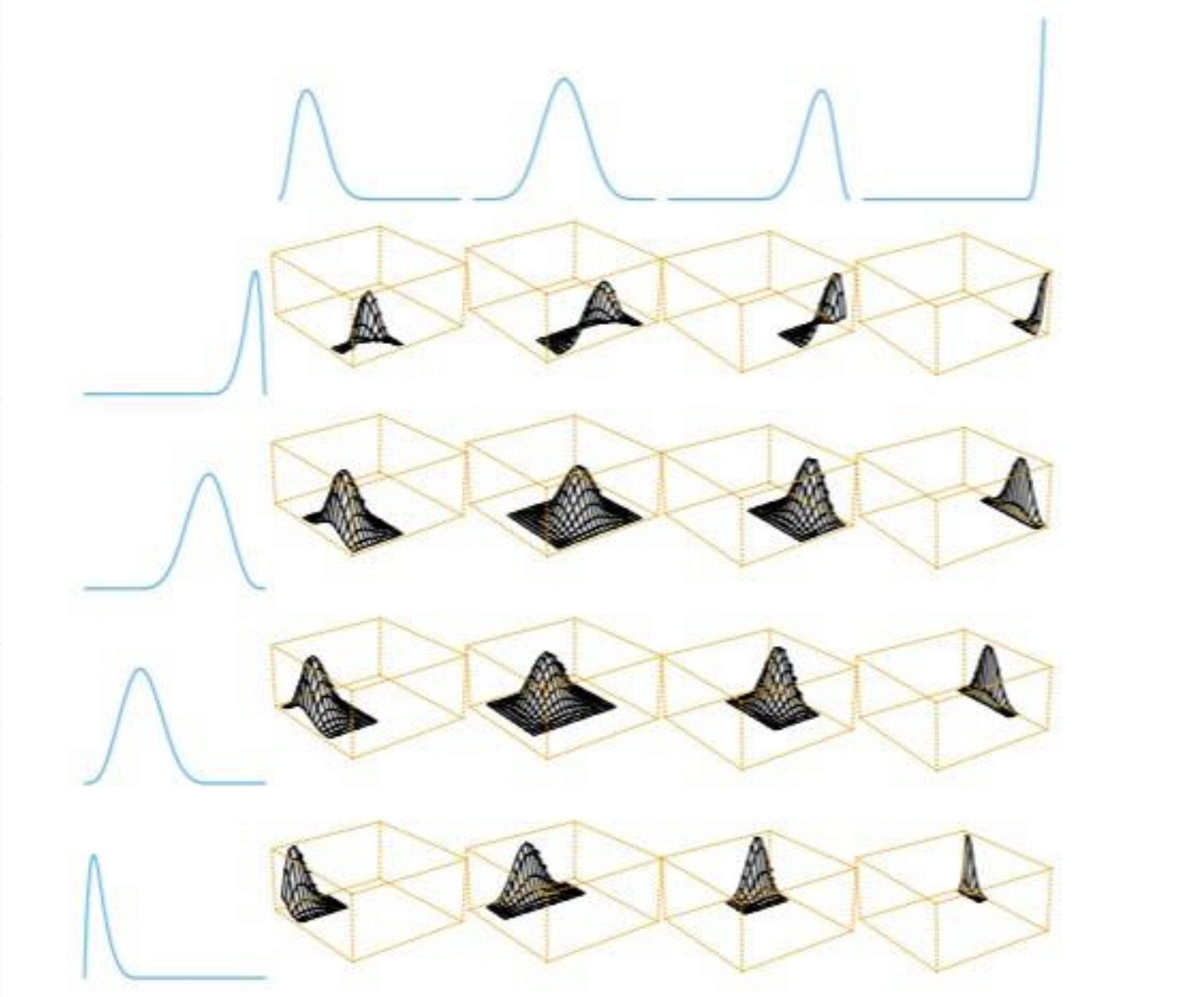


Figure 7: A tensor product basis of B-splines, showing some selected pairs.

B-Splines

- a piecewise polynomial curve on the interval [a, b] has a B-spline basis representation with Bezier curves.
- The i-th B-spline basis function of degree p is denoted by $N_{i,p}(t)$
- defined recursively as

$$N_{i,p}(t) = \begin{cases} 1 & ti \leq t < ti_{+1} \\ 0 & otherwise \end{cases}$$

$$N_{i,p}(t) = \frac{t - t_i}{t_{i+p} - t_i} N_{i,p-1}(t) + \frac{t_{i+p+1} - t}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(t)$$

where $i = 0, \dots, n$ and $p \geq 1$.

- A B-spline curve of degree p with control points $b_0, \dots, bn$ is defined on the interval $[a, b] = [t_m, t_{m\_p}]$ as

$$B(t) = \sum_{i=0}^{n} b_i N_{i,p}(t)$$

Case Study

Dataset:

- clothing sales data of men’s fashion store in Netherland
- 400 observations × 13 columns

Selected Data for Application:

- independent variable: investment in automation
- dependent variable: annual sales in Dutch guilders

Case Study

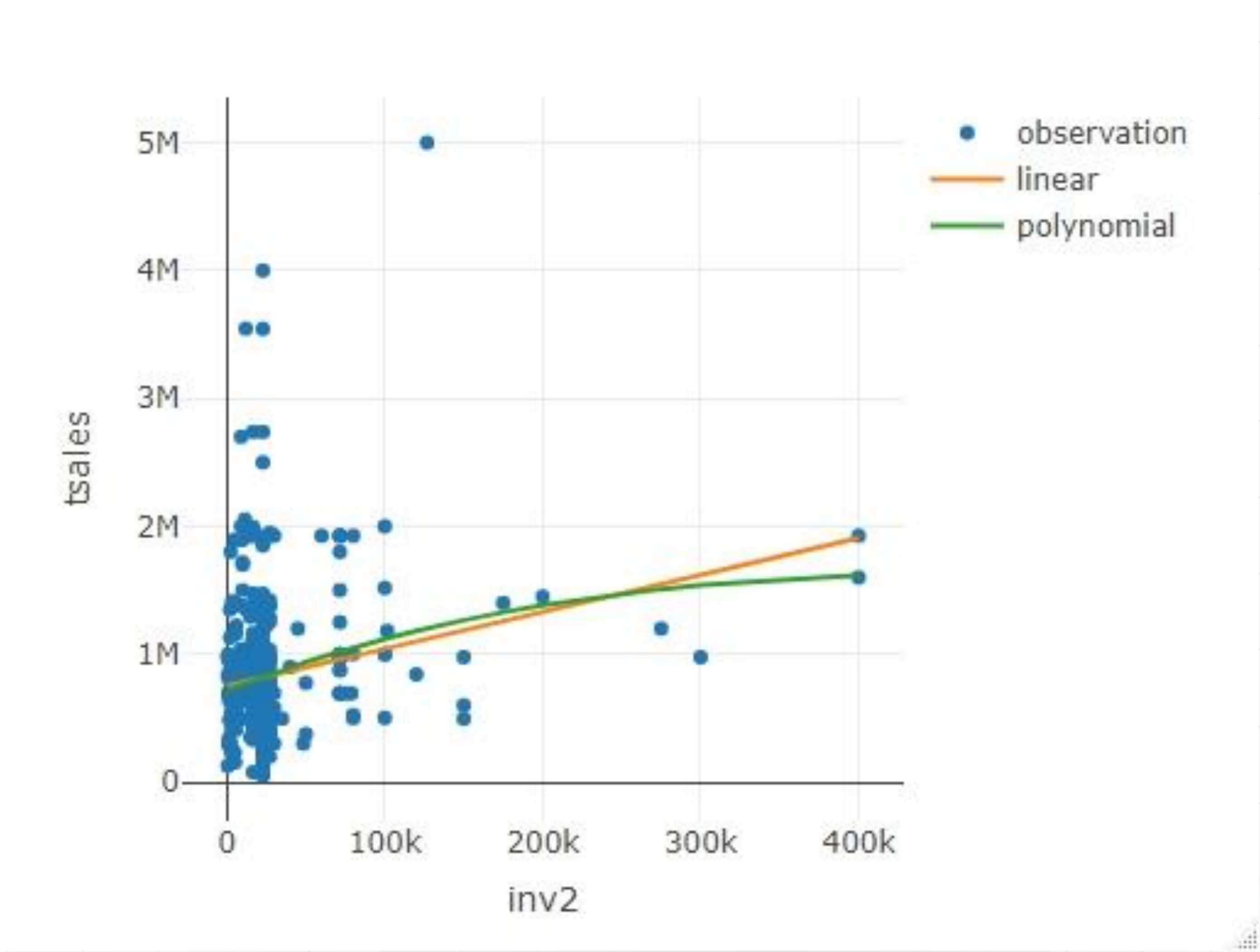


Figure 8: Sample Data Linear and Polynomial Model

Case Study

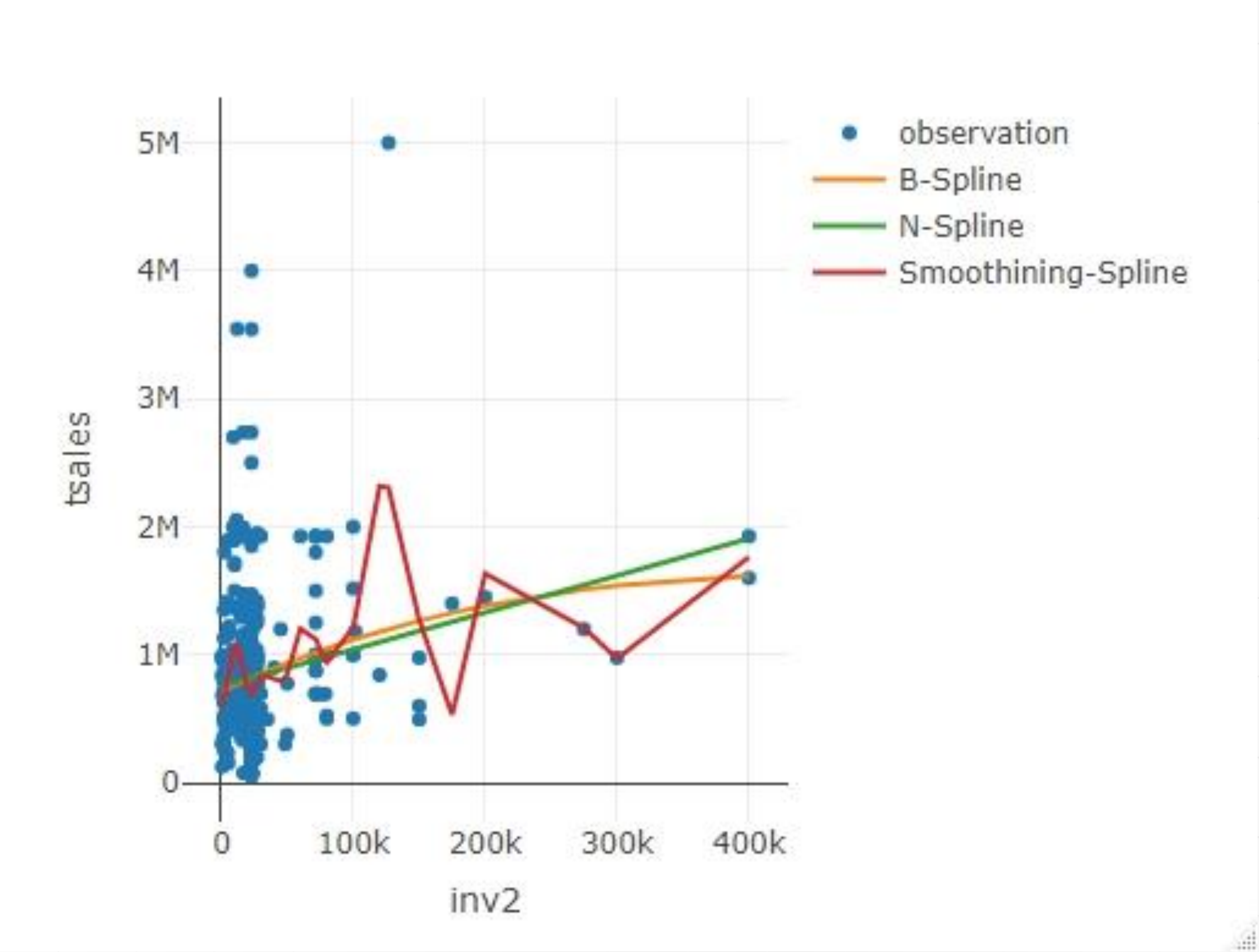


Figure 9: Splines with no specified degree of freedom

Case Study

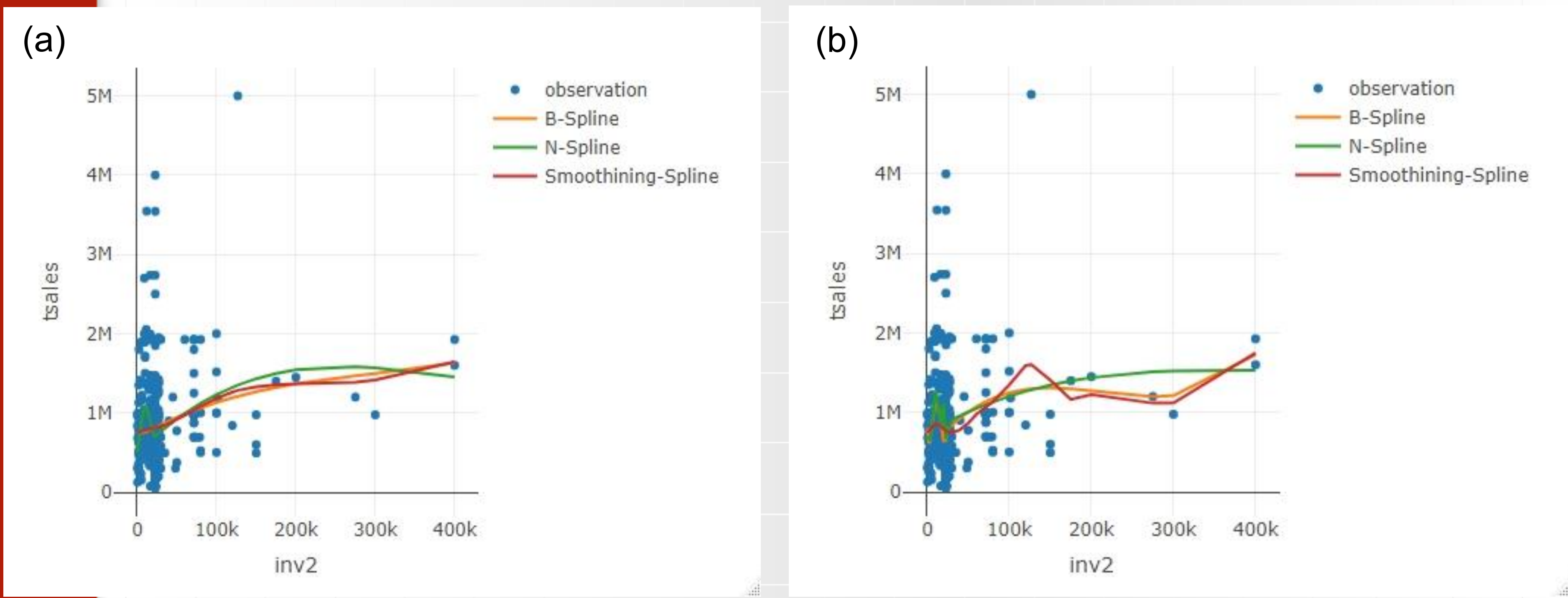


Figure 10: Splines with (a) degree of freedom = 4 and (b) degree of freedom = 8

Case Study

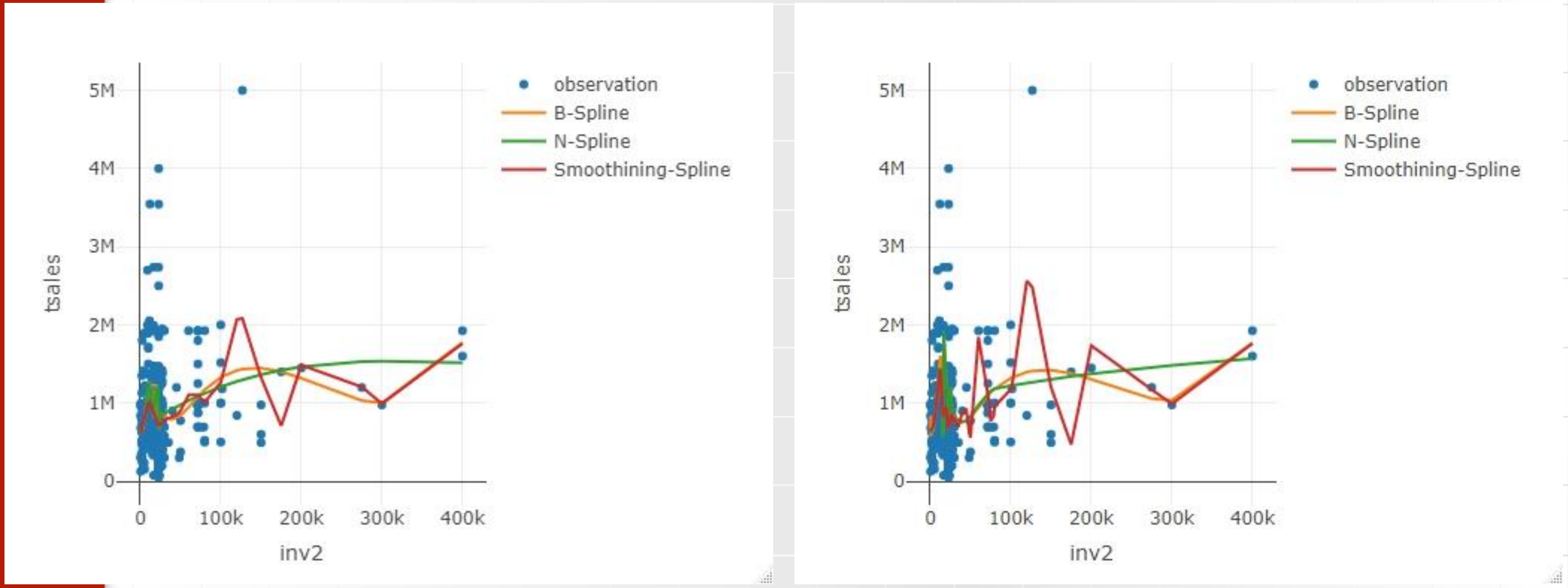


Figure 11: Splines with (a) degree of freedom = 12 and (b) degree of freedom = 24

Case Study

Smoothing Parameter

df_{λ}: 2.791762
λ: 0.1138609
RSS: 29
$No. of\ Iterations$: 29

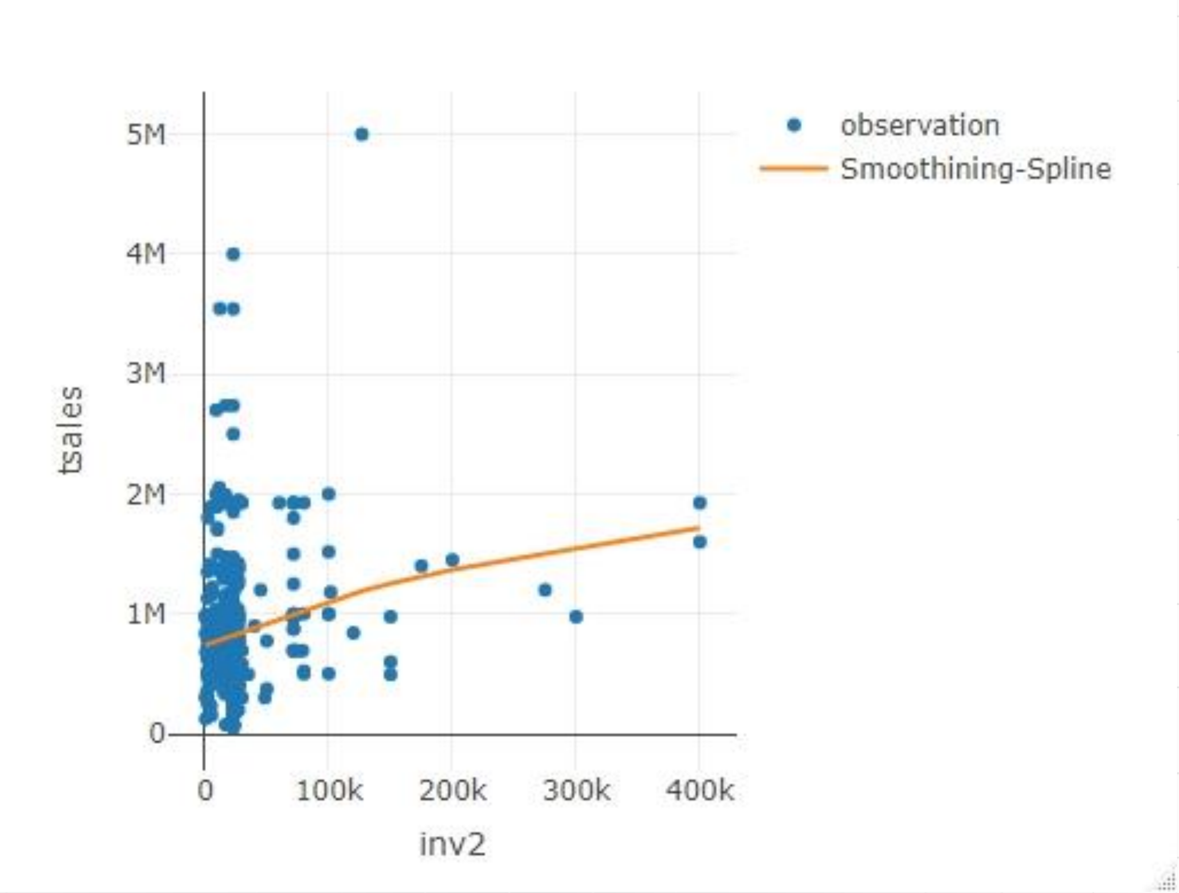


Figure 12: Splines with degree of freedom from iteration

Conclusion

The application of splines in statistical learning is

- ❖ a powerful tool for interpolating purposes
- ❖ helpful in improving a model due to wide range oof parameters
- ❖ useful for generating great fits

However, its disadvantages lies in

- ❖ the choice of parameter which is not always intuitive
- ❖ computational extensiveness
- ❖ the relative difficulty to understand complicated models

Wroclaw University of Science and Technology

Thank you for Listening!

References

- Schoenberg I.J. (1988) Contributions to the Problem of Approximation of Equidistant Data by Analytic Functions. In: de Boor C. (eds) I. J. Schoenberg Selected Papers. Contemporary Mathematicians. Birkhäuser, Boston, MA
- Hastie, Trevor & Tisbshirani, Robert & Friedman, Jerome & Franklin, James (2004). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Math. Intell. 27:(83-85)

R Code for Illustration

```
# SPLINE ILLUSTRATION
age<-c(25,27,28,29,30,33,34,35,36,37,39,41,42,45,50)
salary<-
c(1500,2500,3200,3400,3600,4100,4700,5200,4200,3800,3300,2900,2400,1700,
1500)
datasal<-data.frame(age, salary)

# Linear and Polynomial Fit
fit<-lm(salary~age, data=datasal)
fit2<-lm(salary~poly(age,2)+age, data=datasal)

# Step Function Transformation
datasal$Xbar<-ifelse(datasal$age>35,1,0)
datasal$diff<-datasal$age - 35
datasal$X<-datasal$diff*datasal$Xbar

#Fitting Spline
regsal<-lm(salary~age+X, data=datasal)
```

R Code for Case Study

```
#Libraries For Spline Regression
library(splines)
library(pracma)
library(Ecdat)
data(Clothing)

#linear and polynomial model
lm.parameters<-lm(tsales~inv2)
poly.parameters<-lm(tsales~poly(inv2,3)+inv2)

#splines with no df
fitbs<-lm(tsales~bs(inv2))
fitns<-lm(tsales~ns(inv2))
fitsp<-smooth.spline(tsales~inv2)

#splines with df=4
fitbs<-lm(tsales~bs(inv2, df=4))
fitns<-lm(tsales~ns(inv2, df=4))
fitsp<-smooth.spline(tsales~inv2, df=4)

#Smoothing Splines by Cross Validation
fitsp2<-smooth.spline(x=inv2,y=tsales,cv=TRUE)
```