

1 Practical Considerations

In Chapter ??, we discussed the theoretical basis for shape-constraint P-splines and their ability to incorporate a priori domain knowledge by choice of the constraint term described by the mapping matrix \mathbf{D}_c and the weighting matrix \mathbf{V}_c . We will now consider the practical application of these, as well as their limits in terms of data fitting and constraint holding.

It is important to notice that the addition of the constraint term in (??) further reduces the effective degree of freedom of the model, similar as for P-splines, resulting in a less flexible model. We therefore expect that the measured metric on the training data will be worse compared to a pure B-spline fit. Nevertheless, the metric of interest is the measured error on the validation data, i.e. the held out data that the model has not seen before. Supposing that the a priori domain knowledge reflects the true, underlying function, we expect the measured error on the validation data to be lower than or equal to the error given by an optimal P-spline fit. Here, optimality is based on the optimal smoothing parameter λ given by generalized cross-validation, see Section ?. This can be seen by recognizing the equivalence of the objective functions for P-splines, see (??), and shape-constraint P-splines, see (??), when the underlying B-spline fit does not violate the user-defined constraint, i.e. all $v_j = 0$. This feature is one of the limits of shape-constraint P-splines, i.e. we cannot influence the model using this approach if no constraint violations are present. On the other hand, if the a priori domain knowledge reflects the underlying function, we can expect a far better generalization capability of the model compared to the B-spline fits, especially in situations of noisy or sparse data.

In this chapter, we are going to discuss the practical incorporation of a priori domain knowledge based on the example of peak behavior, see Section 1.1. We will evaluate the performance of shape-constraint P-splines using sparse and noisy data and compare it to B-splines and P-splines. Further, we will discuss the effect of the constraint parameter λ_c , see Section 1.3.

1.1 Peak Constraint in Practice

We will now use shape-constraint P-splines and the a priori domain knowledge of peak behavior to fit the data $D = \{(x^{(i)}, y^{(i)}), i = 1, 2, \dots, n\}$. The data is artificially generated by random sampling of $n = 200$ points $x^{(i)}$ of the function f , i.e.

$$y^{(i)} = f(x^{(i)}) + \epsilon^{(i)} = \exp\left(-\frac{(x^{(i)} - 0.35)^2}{0.1}\right) + \epsilon^{(i)} \quad \text{for } x^{(i)} \in [0.1, 0.8], \quad (1.1)$$

with $\epsilon^{(i)}$ being Gaussian noise with mean $\mu = 0$ and variance $\sigma^2 = 0.01$. We randomly split the data in a training set D_t with 150 samples and a validation set D_v with 50 samples. At

first, we use $d = 45$ B-spline basis functions of order $l = 3$ to fit a B-spline to the training data D_t . Then, we fit a P-spline using the same number of basis functions d and order l with an optimal smoothness parameter $\lambda_{opt} = 7.74$ chosen by generalized cross-validation. Finally, we enforce the peak behavior by using a shape-constraint P-spline using the same number of basis functions d and order l as well as the optimal smoothness parameter $\lambda_{opt} = 3.68$ and the constraint parameter $\lambda_c = 6000$ reflecting high trust in the a priori domain knowledge. The various fits are evaluated on the validation data D_v and shown in Figure 1.1. The mean squared errors on the validation data, as well as on the true, underlying function in (1.1), are given Table 1.1.

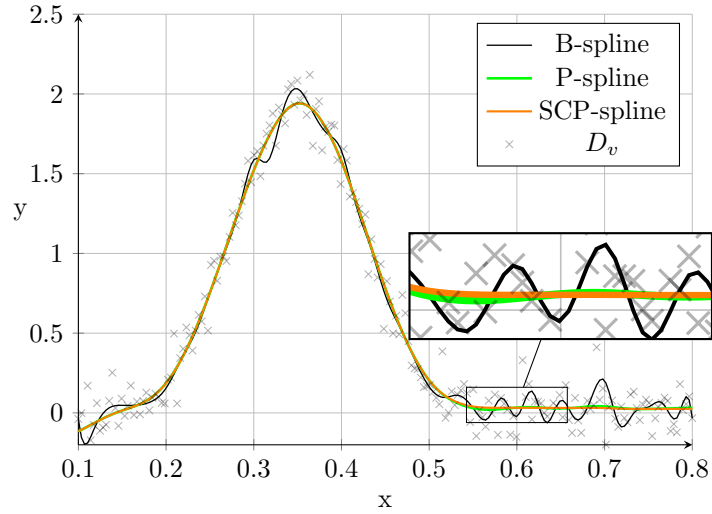


Figure 1.1: B-spline, P-spline and SCP-spline fit for D_v .

The B-spline, as black curve in Figure 1.1, captures the basic shape of the true function, but the flexibility of the B-spline, due to the high number of B-splines, leads to a wiggly curve especially for the almost constant part. This violates the peak behavior of the true function, i.e. being non-increasing after the peak value. For the P-spline, as green curve, this problem relaxes due to the smoothing aspect of the penalty term but does not vanish, as seen in the magnified part in Figure ???. Note that the additional smoothness penalty given by the P-spline already fits the data almost perfectly. The SCP-spline fit adjusts only the parts of the P-spline, which violate the constraint. It may be seen as "fine-tuning" of the fit using the a priori domain knowledge. Hence, it is the best solution here as it is nearly constant for the necessary parts of the function in (1.1).

Model	MSE_{D_v}	$\text{MSE}_{D_{v,true}}$
B-spline	$1.9 \cdot 10^{-2}$	$7.04 \cdot 10^{-3}$
P-spline	$1.22 \cdot 10^{-2}$	$1.52 \cdot 10^{-3}$
SCP-spline	$1.22 \cdot 10^{-2}$	$1.48 \cdot 10^{-3}$

Table 1.1: Mean squared errors on the validation set.

The mean square errors on the noisy validation data D_v in Table 1.1 for P-spline and SCP-spline are almost identical and do not show a favorable model. Comparing the various models with the true, underlying function, see $\text{MSE}_{D_{v,true}}$ in Table 1.1, leads to the assessment that the SCP-spline is the more accurate model with regards to the true function behavior. This coincides with the graphs in Figure 1.1. Hence, the incorporation of a priori domain knowledge via shape-constraints improves the generalization capability measured by the mean squared error on the true function values.

1.2 Sparse Data and the Peak Constraint

We will now examine the behavior of B-, P- and SCP-splines for sparse data, i.e. little data and uneven distributed, sampled from the function in (1.1). The data set D now contains 70 data points distributed in a way that there is little data in the peak region, i.e for $x \in [0.2, 0.5]$ we have only 10 data points. We perform a random train-validation split of the data in D , i.e. the training data D_t consists of 52 points and the validation data D_v consists of 18 points. We follow the same approach as above, i.e. fit a B-spline, perform generalized cross-validation to fit the optimal P-spline and finally apply the shape-constraint to enforce peak behavior. The small data set indicates that a different, not equidistant knot placement may be helpful. Hence, we carry out 2 experiments, one with equidistant knot placement and the other with quantile-based knot placement, see Section ???. We chose to use $d = 25$ B-spline basis functions of order $l = 3$. The optimal smoothness parameter was given as $\lambda_c = 0.00657$ for the equidistant fit and $\lambda_{opt} = 0.00215$ for the quantile-based fit. The constraint parameter λ_c was set to $\lambda_c = 1000$ for both fits, reflecting high trust in the a priori domain knowledge. The resulting fits are shown in Figure 1.2 and Figure 1.3.

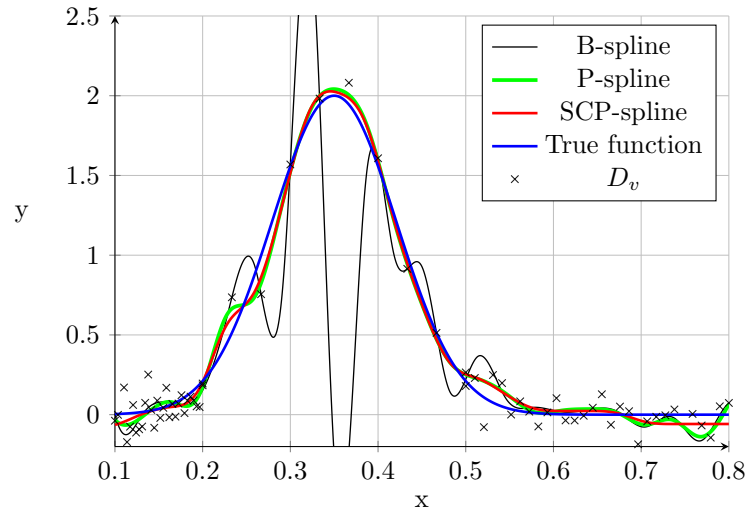


Figure 1.2: Equidistant B-spline, P-spline and SCP-spline fit for sparse data D .

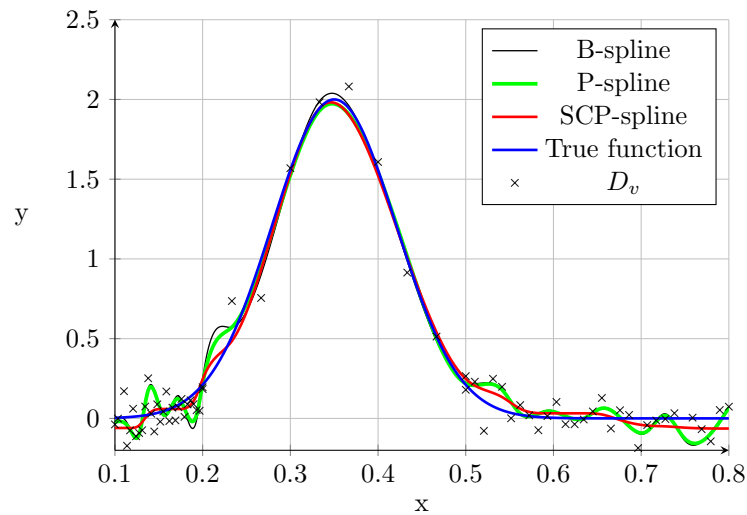


Figure 1.3: Quantile-based B-spline, P-spline and SCP-spline fit for sparse data D .

1.3 Lambda c

1.4 Something

In this chapter we use the theory discussed in Chapter Chapter ?? to estimate uni- and bivariate functions using data and a priori domain knowledge. An overview of the different problems considered in this chapter is given in Table Table 1.2.

Univariate	Section	Bivariate	Section
B-splines		Tensor-product B-splines	
P-splines		Tensor-product P-splines	
SCP-splines		Tensor-product SCP-splines	

Table 1.2: Problem overview.

First, we are using B-splines, see Section ??, for the estimation of the unknown function $y = f(x)$, i.e. we solve the optimization problem

$$\arg \min_{\beta} Q_1(\mathbf{y}, \beta) = \|\mathbf{y} - \mathbf{X}\beta\|, \quad (1.2)$$

using the B-spline or tensor-product B-spline basis matrix \mathbf{X} . Next, we use the concept of P-splines, see Section ??, to estimate smooth functions, i.e. we solve the optimization problem

$$\arg \min_{\beta} Q_2(\mathbf{y}, \beta; \lambda) = \|\mathbf{y} - \mathbf{X}\beta\| + \lambda \cdot \text{pen}(\beta), \quad (1.3)$$

where $\text{pen}(\beta)$ specifies a smoothness penalty term. Finally, we are going to incorporate a priori domain knowledge into the fitting process using shape-constrained P-splines (SCP-splines), i.e. we solve the optimization problem

$$\arg \min_{\beta} Q_3(\mathbf{y}, \beta; \lambda, \lambda_c) = \|\mathbf{y} - \mathbf{X}\beta\| + \lambda \cdot \text{pen}(\beta) + \lambda_c \cdot \text{con}(\beta), \quad (1.4)$$

where $\text{pen}(\beta)$ is again a smoothness penalty term and $\text{con}(\beta)$ specifies the user-defined shape-constraint to incorporate a priori domain knowledge with, see [hofner2011monotonicity] and [bollaerts2006simple]. Various types a priori domain knowledge can be incorporated using the constraints listed in Table ??.

The focus of this chapter is the definition and use of shape-constraint P-splines, which are characterized by their parameters β given by solving the optimization problem (1.4).

Eidesstattliche Erklärung

Hiermit erkläre ich, dass die vorliegende Arbeit gemäß dem Code of Conduct – Regeln zur Sicherung guter wissenschaftlicher Praxis (in der aktuellen Fassung des jeweiligen Mitteilungsblattes der TU Wien), insbesondere ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel, angefertigt wurde. Die aus anderen Quellen direkt oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder in ähnlicher Form in anderen Prüfungsverfahren vorgelegt.

Wien, XX. Monat, Jahr

Vorname Nachname