

Chapter 4 - Experiments

Weber Jakob

October 16, 2020

Contents

1	Overview	1
2	Experiments	3
2.1	Exp. 1.1: Static Function Approx.using a priori Knowledge on Function 1	3
2.2	Exp. 1.2: Static Function Approx. using a priori Knowledge on Function 2	4
2.3	Constraint vs. Noise	5
2.3.1	Exp. 2: Different Noise Levels	5
2.3.2	Exp. 3: Different Noise Colors	6
2.4	Well-distributed vs. Skewed Data and Knot Placement	7
2.4.1	Exp. 4: Equidistant vs. Quantile Based Knot Placement for Grid Data	8
2.4.2	Exp. 5: Equidistant vs. Quantile Based Knot Placement for Skewed Data	9
3	Exp. 6: Real life data	9

1 Overview

Following the descriptions given in the previous chapters, we are now going to test the algorithm of structured additive regression using a priori knowledge on the problems given in Table 1. We will use two different, artificial functions with known behavior and one set of real life data, for which the physical a priori knowledge is given.

Exp.	Problem Definition	Function	n_{data}	k
1.1	Static function approx. using a priori knowledge	1	100	35
1.2	Static function approx. using a priori knowledge	2	500	35
2	Different noise levels	2	150	35
3	Different noise colors	1	200	35
4	Well-distributed data and knot placement	2	250	35
5	Skewed data distribution and knot placement	2	250	35
6	Real life data	-	-	-

Table 1

The first function is given by

$$f(x) = \begin{cases} 0 & \text{if } x \leq 0.8 \\ 20 \sin(x - 1.2) & \text{else} \end{cases}, \quad \text{for } x \in [0, 2.5]. \quad (1)$$

It is a constant function till $x = 0.8$ and afterwards increasing. An example of the true function as well as noisy samples from it is given in Fig. 1.

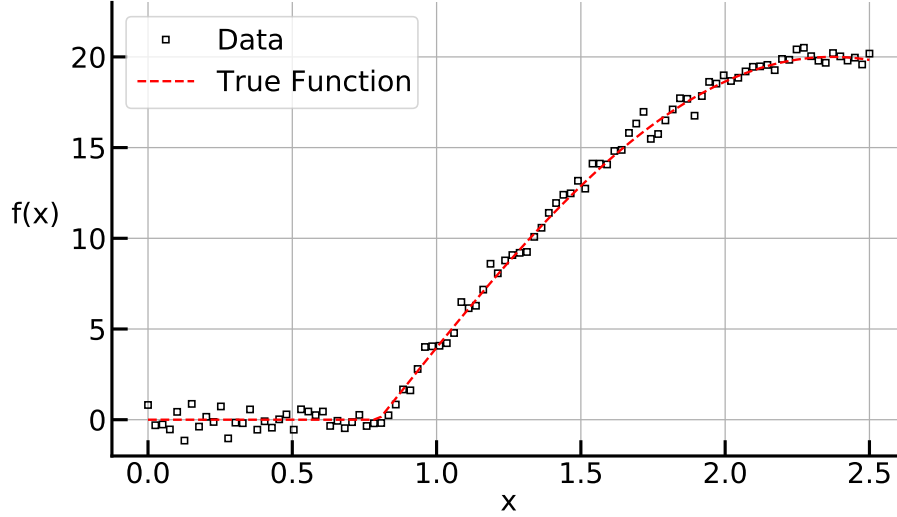


Figure 1: Test function 1

The second function that will be investigated is given by

$$f(x) = \begin{cases} 2 \exp\left(-\frac{(x-0.5)^2}{0.05}\right) + 3x & , \text{if } x \leq 0.85 \\ 2 \exp\left(-\frac{(0.85-0.5)^2}{0.05}\right) + 3 * 0.85 & , \text{else} \end{cases}, \quad \text{for } x \in [0, 1]. \quad (2)$$

The true functions depicts a unimodal behavior and is constant for $x > 0.85$. To test the algorithm, we deliberately generate "wrong" data using the following function

$$f(x) = 2 \exp\left(-\frac{(x-0.5)^2}{0.05}\right) + 3x \quad \text{for } x \in [0, 1]. \quad (3)$$

An example of the true function 2 as well as noisy samples from function 3 are given in the figure 2.

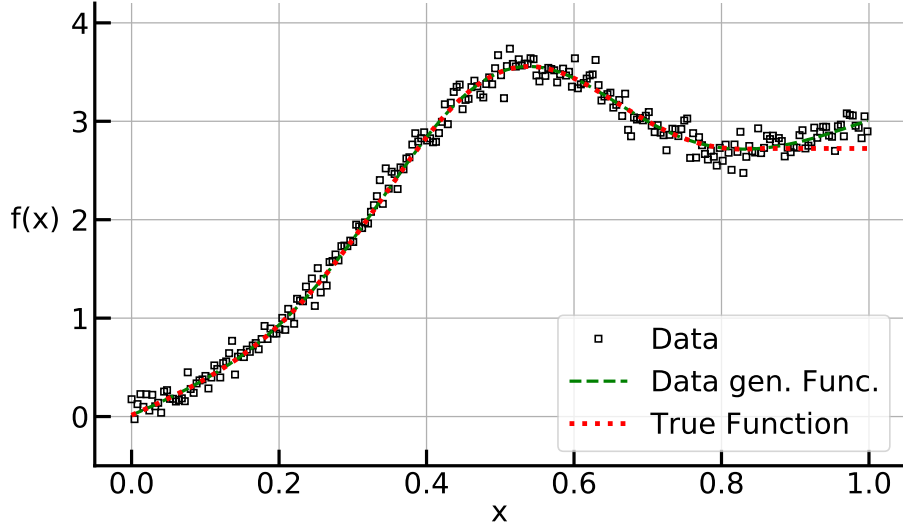


Figure 2: Test function 2

2 Experiments

2.1 Exp. 1.1: Static Function Approx.using a priori Knowledge on Function 1

For the experiment using function 1, the data set contains 100 points. We use $k = 35$ as number of splines. The smoothing parameter λ_s is optimized using cross-validation given in Chapter *CrossValidation*. The parameter λ_c regulating the influence of the constraint is set to the 1000-fold of the smoothing parameter λ_s . The resulting constrained fit, and for comparison, the unconstrained fit using the optimal smoothing parameter $\lambda_s = 0.1615$ are shown in the following figure.

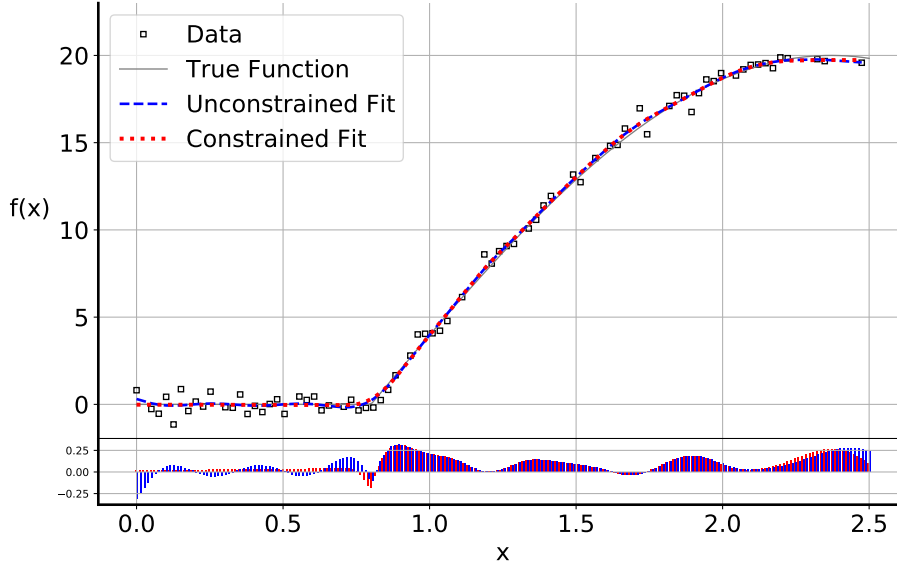


Figure 3: Constr. fit, unconstr. fit and residuals (lower plot) for function 1

The red line follows the a priori knowledge of increasing behavior better than the blue, unconstrained line. Especially in the constant area, e.g. $x \in [0, 0.8]$, the constraint improves the static function approximation and lowers the residual as seen in the lower part of figure 3. In the area of increasing behavior, the two fits are almost identical since the constraint is inactive. The mean squared error on the training data, on the test data and on the true function are given in the Table *Results*.

2.2 Exp. 1.2: Static Function Approx. using a priori Knowledge on Function 2

For the test using function 2 the data set contains 500 points. There are two possibilities to include the a priori knowledge of a unimodal behavior. First we can use the peak constraint matrix given in Chapter *Penalty Matrices*. On the other hand, we can use the concavity constraint, since concave functions are likely to have a peak when the data shows a peak. We examine both possibilities in Figure 4. For each fit, we used $k = 35$ as number of splines and the smoothing parameter $\lambda_s = 0.6189$ was again optimized using cross-validation. The constraint parameter λ_c was set to 618.9.

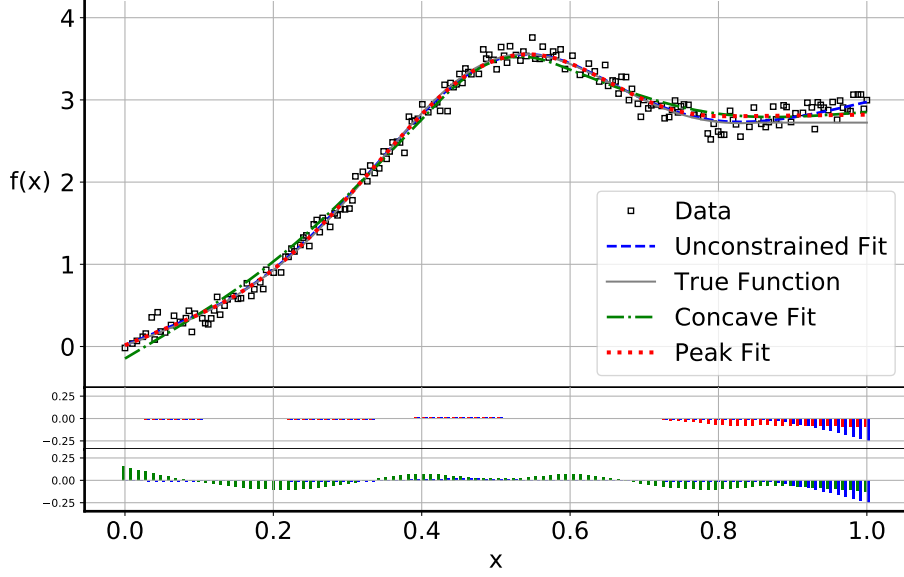


Figure 4: Constr. fit, unconstr. fit and residuals (lower plots) for function 2

The estimate for the concave constraint shows a the desired peak behavior, and is more smooth than the peak constraint estimate. This smoothness leads to changes in the sign of the residuals compared to the unconstrained fit as seen in the lower part of figure 4. The peak constraint identifies the peak correctly and produces a sufficiently smooth fit with low residual. Especially for $x > 0.8$, i.e. the region where the data is on purpose "wrong", the residual of the peak constraint fit is lower than the unconstrained ones. The mean squared errors are given in the *results table*. It is important to notice that the concavity constraint must be used carefully, since not all functions with a peak are also concave or vice versa.

2.3 Constraint vs. Noise

In this section, we investigate the effect of different noise levels as well as noise types on the fitting process. In the first part, various levels of Gaussian noise are applied to function 2. In the second part, we will use different colors of noise acting on function 1. For both examples we use $k = 35$ splines and optimize the smoothing parameter λ_s using cross-validation.

2.3.1 Exp. 2: Different Noise Levels

Gaussian noise is characterized by the following two parameters:

- Location: The mean value μ of the Gaussian distribution.
- Scale: The variance value σ^2 of the Gaussian distribution.

We set the location equal to zero and vary the scale value. The effect of the noise on the fitting procedure is shown in the following figure. The chosen noise levels are $\sigma^2 = \{0.01, 0.05, 0.1\}$.

For these values of σ^2 , the correspond optimized smoothing parameters are $\lambda_{0.01} = 0.3162$, $\lambda_{0.05} = 2.3713$ and $\lambda_{0.1} = 1.211$. The constraint parameter λ_c was set to $\lambda_c = 1000\lambda_s$.

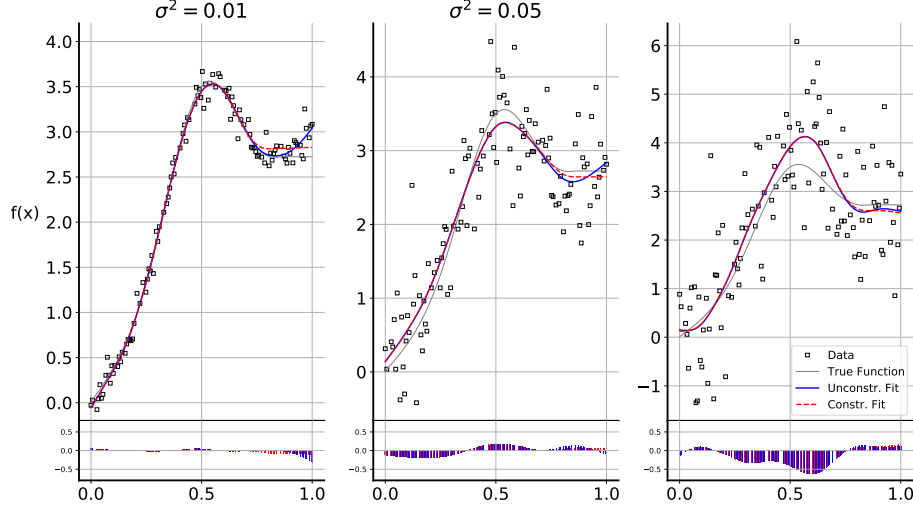


Figure 5: Constr. fit, unconstr. fit and residuals for various noise levels

The left plot shows the fit for $\sigma^2 = 0.01$. The noise level is moderate, which leads to a smooth and satisfying fit. The peak constraint is well satisfied.

The noise level in the middle plot is already high. The unconstrained fit is nevertheless smooth, but it underestimates the peak and violates the constraint for $x > 0.9$. The constrained fit holds the peak constraint and generalizes better to the test set, as seen in *Table Results MSEs*.

For the right plot and $\sigma^2 = 0.1$, both constrained and unconstrained fit are smooth but overestimate the fit especially in the peak region. The noise level is very high and the fits must be used carefully.

2.3.2 Exp. 3: Different Noise Colors

We investigate the effect of different noise color on the function fitting process. The tested colors are the following power-law noise types:

- White Noise
- Pink Noise
- Brownian Noise

The unconstrained as well as the increasing constrained fit for all noise colors are shown in Figure 6.

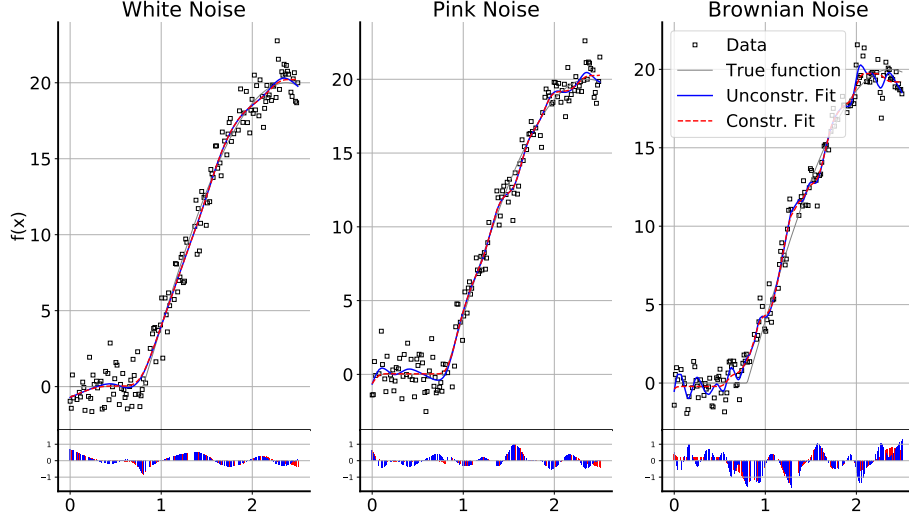


Figure 6: Constr. and unconstr. fit and residuals for noise colors

For the white noise data in the left plot, the increasing constraint produces a good fit according to the constraint and the residual except for low $x < 0.2$. The optimized smoothing parameter for the unconstrained fit was given by $\lambda_s = 9.08$ and the constraint parameter was set to $\lambda_c = 9080$. The fit recovers the true function quite well.

For the pink noise data in the middle plot, the constraint helps regularizing the fit, but the increasing constraint is not satisfied for low $x < 0.01$. The increasing part of the function, i.e. $f(x)$ for $x \geq 1.2$, was fit well. The smoothing parameter was optimized using cross-validation and set to $\lambda_s = 0.3$. The constraint parameter was set to $\lambda_c = 6000$.

For the brownian noise data in the right plot, the increasing part of the function was recovered well, but for $x \geq 2$, the fit diverges the true function because of the strong noise component. The increasing constraint is hold quite well. The optimized smoothing parameter used was $\lambda_s = 0.0056$ and the constraint parameter was set to $\lambda_c = 6000$.

2.4 Well-distributed vs. Skewed Data and Knot Placement

We will now investigate the effect of irregular data distributions on the fitting process. We therefore examine input data $x^{(i)}$ for $i = 1, \dots, n$ generated on a grid vs. data generated from a beta distribution given by

$$f(x) = \frac{1}{B(a, b)} x^{a-1} (1-x)^{b-1} \quad (4)$$

with

$$B(a, b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)} = \int_0^1 u^{a-1} (1-u)^{b-1} du$$

for different values of the parameters a and b . We will investigate the effect of the knot placement, which can either be equidistant knot placement or quantile-based knot placement, using this data. We again use noisy samples from function 2.

2.4.1 Exp. 4: Equidistant vs. Quantile Based Knot Placement for Grid Data

We will use 250 noisy samples from function (1) generated from a linear sequence of x . The smoothing parameters for the equidistant knot placement fit and for the quantile based knot placement fit were optimized using cross-validation and set to $\lambda_{s, \text{equidistant}} = 1.211$ and $\lambda_{s, \text{quantile}} = 4.641$. The constraint parameter for the peak constraint was set to $\lambda_c = 6000$ for both fits. We use 35 splines for both fits. The resulting fits are shown in figure 7.

As expected for grid data, there is not much difference in the fits between the knot placement types. The peak constraint is hold by both fits quite well. The residuals for equidistant knot placement are overall lower than for quantile-based knot placement. It is important to notice that the discontinuities in first derivative of the spline basis for both fits is just visual and comes from the number of used data points for the plot.

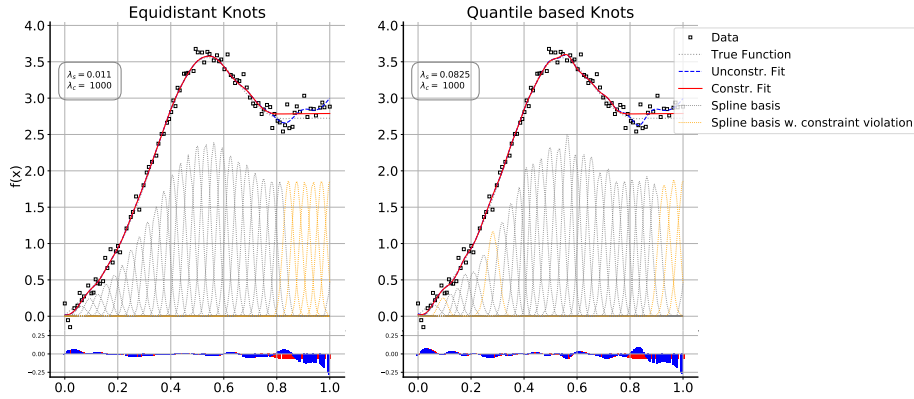


Figure 7: Constr. and unconstr. fit and residuals for grid data

2.4.2 Exp. 5: Equidistant vs. Quantile Based Knot Placement for Skewed Data

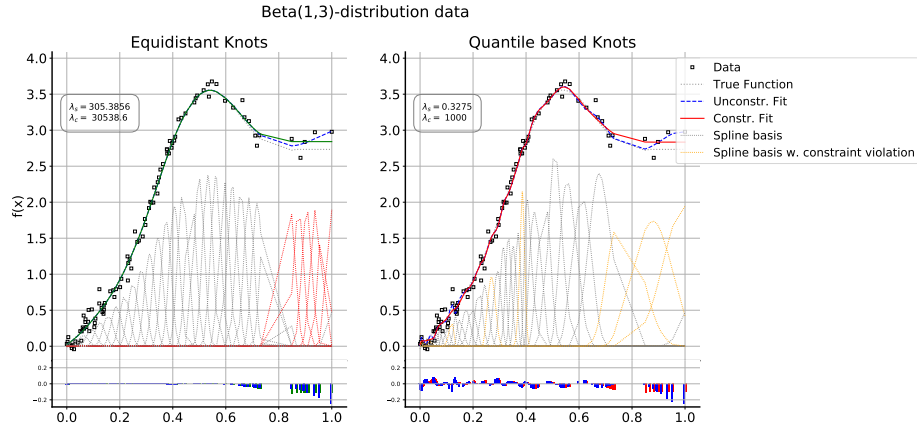


Figure 8: Constr. and unconstr. fit and residuals for left skewed data

3 Exp. 6: Real life data

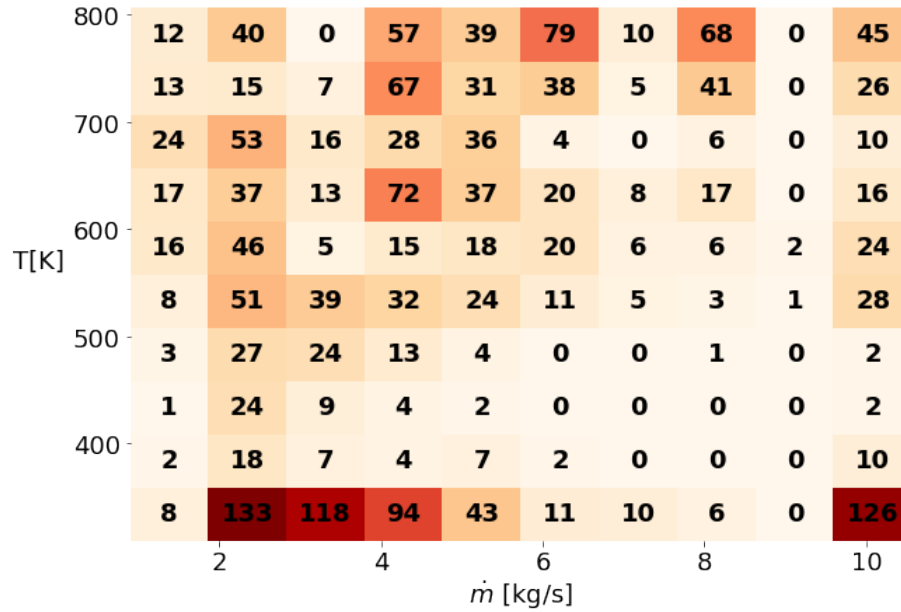


Figure 9: Data Situation

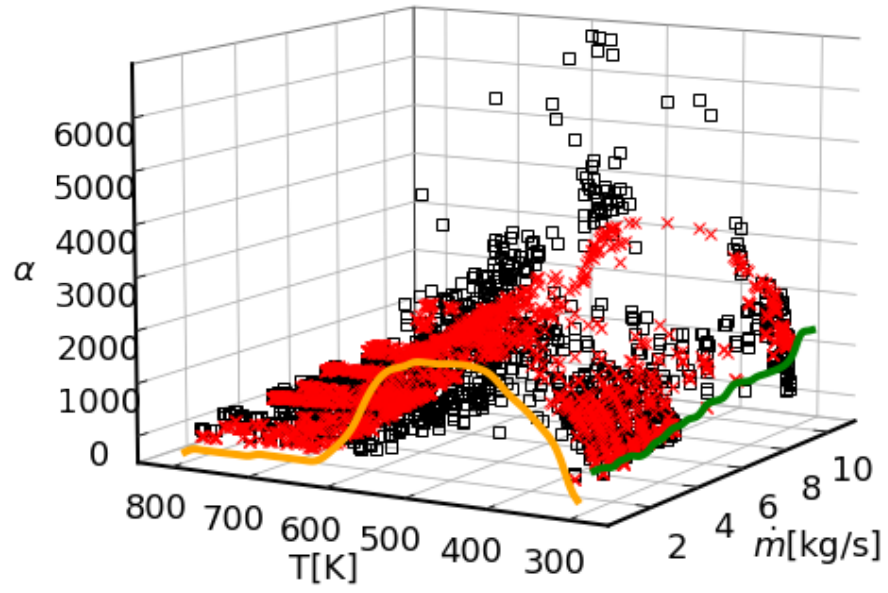


Figure 10: Constr. and unconstr. fit for Ebner Data