

1 **Contents**

2	Notation	ii
3	1 Introduction	1
4	1.1 XXX motivation - why is it important	1
5	1.2 XXX problembaum / fragestellungen	1
6	1.3 XXX State-of-the-art	1
7	1.4 Roadmap	1
8	2 Problem Description	2
9	2.1 Available Data	2
10	2.1.1 Sentinel 2 Satellite Image Data	2
11	2.1.2 Yieldmapping Data	4
12	2.1.3 Gather Data	4
13	3 Interpolation Methods	7
14	3.1 Setting	7
15	3.2 XXX DAS vs GDD	7
16	3.3 Robustify	7
17	3.3.1 XXX Our Adjustment:	9
18	3.4 Parametric Regression	9
19	3.4.1 Double Logistic	9
20	3.4.2 Fourier Approximation	10
21	3.5 Non-Parametric Regression	10
22	3.5.1 Kernel Regression	11
23	3.5.2 Kriging	11
24	3.5.3 Savitzky-Golay Filter (SG Filter)	12
25	3.5.4 Locally Weighted Regression (LOESS)	14
26	3.5.5 B-splines	15
27	3.5.6 Natural Smoothing Splines	16
28	3.5.7 XXX Whittaker Smoother	16
29	3.6 Tuning parameter estimation	16
30	3.7 Robustification – Recap	17
31	3.7.1 Upper Envelope Approach - Penalty for negative resiudals	17
32	3.8 Performance Assecement	17
33	4 NDVI Correction / Improve NDVI Data	20
34	4.1 Considering other SCL Classes	20
35	4.2 XXX Correction	21
36	4.2.1 XXX idea -and- stepwise plots	21
37	4.2.2 XXX data-table-construction	21
38	4.2.3 XXX ml-methods	21
39	4.2.4 XXX Uncertainty	25
40	4.3 XXX Evaluation Method	25
41	4.3.1 yield estimation	26
42	5 Results	27
43	5.1 XXX small recap from “Interpolation Methods”	27

44	5.2 Robustification and NDVI-Correction	27
45	6 Discussion	28
46	7 Outlook	29
47	7.1 Data	29
48	7.2 Interpolation	29
49	7.3 NDVI Correction	29
50	7.4 NDVI Correction + +	29
51	Bibliography	30
52	A XXX Appendix	32

53 Notation

- 54 c : a (vector of) constant(s)
- 55 $\lambda \in \mathbb{R}$: a scalar
- 56 $n \in \mathcal{N}$: sample size
- 57 i, j are indices in $\{1, \dots, n\}$
- 58 $x \in \mathbb{R}^n$: covariate in 1-dim interpolation setting
- 59 $w \in \mathbb{R}^n$: a vector of weights for each location x
- 60 $y \in \mathbb{R}^n$: response in 1-dim interpolation setting
- 61 $\hat{y} \in \mathbb{R}^n$: estimate of y
- 62 $r \in \mathbb{R}^n$: residuals given by $y - \hat{y}$
- 63 Pixel: A pixel describes a specific location in a field. It has the size of 10 x 10 meters
64 and coincides with the resolution (and location) of the sentinel-2 pixels. Such pixels are
65 illustrated in figure ???. Additional information like yield is also attached.
- 66 P_t : this describes the observed data (weather and spectral bands) at time t and the location
67 of one pixel.
- 68 P : a pixel. We see it as a collection of all the observations at the specified location within
69 one season. More formally, $P := \{P_t | t \text{ is a valid sample time within a defined season}\}$
- 70 SCL: scene classification layer. This indicates what one can expect at a pixel at a sampled
71 time. For an overview cf. table 2.2
- 72 P^{SCL45} : similar to P but we only consider observations which belong to the classes 4 and
73 5. This is used done to get a subset of observations which are less contaminated by clouds
74 and shadows.
- 75 NDVI: normalized vegetation difference index
- 76 DAS: days after sowing
- 77 GDD: growing degree days – cumulative sum of (temperature – threshold)⁺

78 **Chapter 1**

79 **Introduction**

80 **1.1 XXX motivation - why is it important**

- 81 - NDVI-timeseries is very simple and widely used. Examples are: - Plant Models REF -
- 82 Season Start (start of spring) (community name: land-surface-plant-phenology) -
- 83 Since satellite images are “for free” researchers extract

84 **1.2 XXX problebaum / fragestellungen**

85 problem schilderung anhand des Leitfadens: **pictures?**

86 **1.3 XXX State-of-the-art**

- 87 zusammenfassung mit literaturrecherche hier:
- 88 — Doublelogistic (winter-ndvi)
- 89 — parametric / non-parametric approaches
- 90 — spatio-temporal approaches

91 **1.4 Roadmap**

92 In chapter

93 **Chapter 2**

94 **Problem Description**

95 **2.1 Available Data**

96 Our study region is a farm of over 800ha, which is located in western Switzerland. From
97 REF-gregor we acquire satellite image data (section 2.1.1), yield maps of several cereals
98 from 2017 to 2021 (section 2.1.2), and meteorological data (section 2.1.3).

99 **2.1.1 Sentinel 2 Satellite Image Data**

100 **General Information**

101 The European Space Agency (ESA) ¹ freely distributes the high quality images of the two
102 Sentinel satellites 2 (S2). Together, both satellites have a revisit time of 5 days at the
103 equator and 2-3 at mid-latitudes. However, at our study region we only receive an image
104 every 5 days. In order to decrease the effect of atmospheric conditions like reflections
105 and scattering, we will not work with the raw data but with the results of the Level-2A
106 processing²³.

107 **Data Description**

108 The Level-2A processed images we use contain 12 spectral bands with local resolutions up
109 to 10 meters (see 2.1). Bands which have a lower resolution (20 and 60 meters) will be
110 scaled up to 10 meters using cubic interpolation (REF gregor perich). Additional to the
111 spectral bands the ESA also supplies a Scene Classification Layer (*SCL*) where for each
112 location the observed subject is assigned to an *SCL-class* (cf. table 2.2). In chapter 3 we
113 will use this classification to filter out unreliable data points considering only SCL-classes
114 4 and 5.

115 **Data Illustration**

116 The figure 2.1 shows a selection of 6 satellite images of a field, which display our challenges.
117 In February (image(a)), as expected, we see no vegetation but bare soil. At the beginning

¹REF: <https://sentinel.esa.int/web/sentinel/missions/sentinel-2>

²REF <https://sentinels.copernicus.eu/web/sentinel/technical-guides/sentinel-2-msi/level-2a/algorithms>

³XXXREF gregor perich “Data prior to March 2018 was only available in the top-of-atmosphere L1C format and was downloaded as such [...] L1C data was processed to L2A product level using the ‘Sen2Cor’ processor provided by ESA”

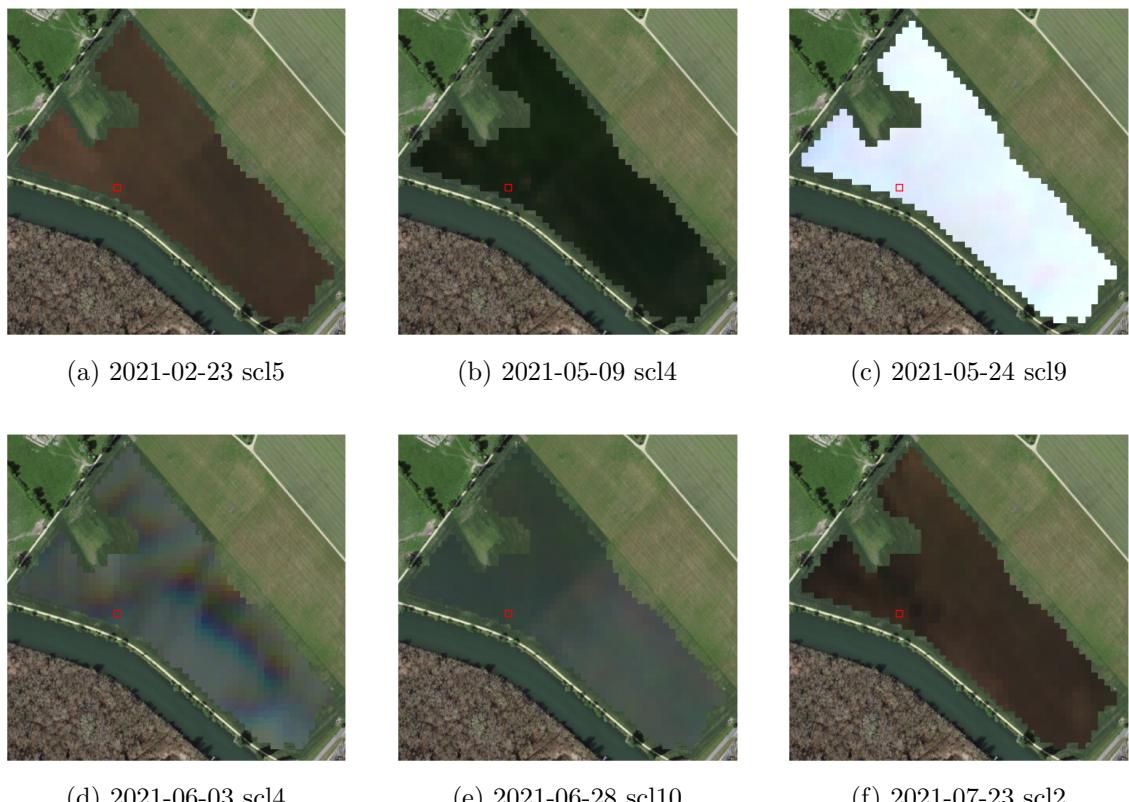


Figure 2.1: Satellite images of a field at selected times with a static background for orientation. The SCL-class of the highlighted pixel is provided in the respective subtitle. (???xxx include scl legend?)

Table 2.1: Jaramaz, Perović, Belanovic Simic, Saljnikov, Cakmak, Mrvić, and Zivotic (Jaramaz et al.) List of spectral bands of the S2-satellites. Each band has its center at the wavelength λ in nm with the spectral width $\Delta\lambda$ in nm with a spatial resolution SR in m.

Band	λ	$\Delta\lambda$	SR	Purpose
1	443	20	60	Atmospheric correction (aerosol scattering)
2	490	65	10	Sensitive to vegetation senescing, carotenoid, browning and soil background; atmospheric correction (aerosol scattering)
3	560	35	10	Green peak, sensitive to total chlorophyll in vegetation
4	665	30	10	Maximum chlorophyll absorption
5	705	15	20	Position of red edge; consolidation of atmospheric corrections / fluorescence baseline.
6	740	15	20	Position of red edge, atmospheric correction, retrieval of aerosol load.
7	783	20	20	Leaf Area Index (LAI), edge of the Near-Infrared (NIR) plateau.
8	842	115	10	LAI
8a	865	20	20	NIR plateau, sensitive to total chlorophyll, biomass, LAI and protein; water vapor absorption reference; retrieval of aerosol load and type.
9	945	20	60	Water vapor absorption, atmospheric correction.
10	1375	30	60	Detection of thin cirrus for atmospheric correction.
11	1610	90	20	Sensitive to lignin, starch and forest above ground biomass. Snow/ice/-cloud separation.
12	2190	180	20	Assessment of Mediterranean vegetation conditions. Distinction of clay soils for the monitoring of soil erosion. Distinction between live biomass, dead biomass and soil, e.g. for burn scars mapping.

118 of May we observe a cloudless dark green field. In (c) it is obvious that we have no chance
 119 to get useful information when there is a heavy cloud cover. Figure (d) shows that the
 120 SCL classification is not reliable, since we evidently observe clouds. In (e) we see a pale
 121 green. This likely shimmers through cirrus clouds.

122 2.1.2 Yieldmapping Data

123 The crop yield data were collected using a combine harvester. Equipped with GPS, the
 124 harvester drives over the fields and continuously estimates the crop density in t/ha (see fig.
 125 2.2a). We take the data set derived from this in REF-Gregor-Perich, where error-prone
 126 measurement points (such as during an egen curve) were removed and then the yield map
 127 was rasterized using linear interpolation (cf. fig. 2.2b).

128 Comparing the manually weighted yield and the sum of estimated raster (per field per
 129 year) we note a discrepancy of about 10% (cf. REF-gregor). Since the relative estimation
 130 error is rather constant and we do not aim to estimate the absolute yield we will not
 131 consider this deviation.

132 2.1.3 Gather Data

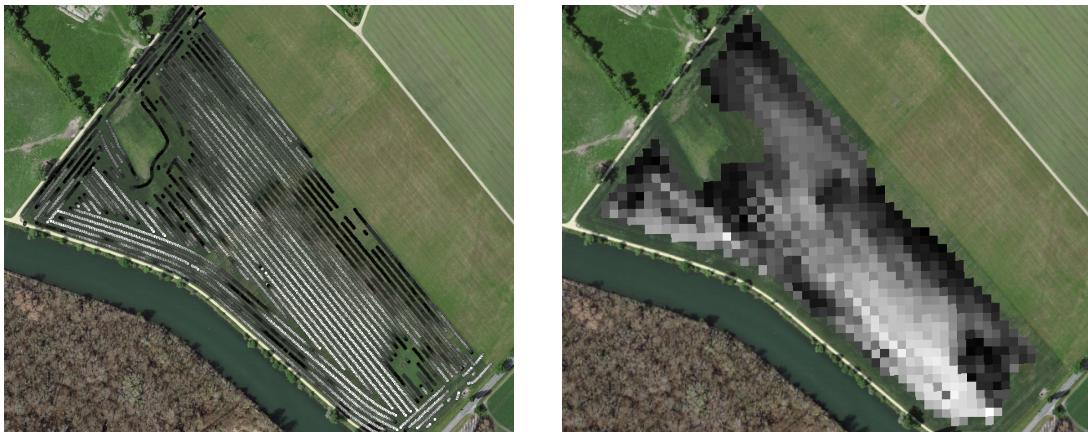
133 Before we join all the data, we define a few concepts.

134 Using bands $B4$ and $B8$, we calculate the well-known Normalized Difference Vegetation
 135 Index ($NDVI$) using the formula: (???REF nötig?)

$$NDVI = \frac{B8 - B4}{B8 + B4} \quad (2.1.3.1)$$

Table 2.2: Overview: Scene Classification Layers (SCL)

No.	Class	Color
0	No Data (Missing data on projected tiles) (black)	
1	Saturated or defective pixel (red)	
2	Dark features / Shadows (very dark gray)	
3	Cloud shadows (dark brown)	
4	Vegetation (green)	
5	Bare soils / deserts (dark yellow)	
6	Water (dark and bright) (blue)	
7	Cloud low probability (dark gray)	
8	Cloud medium probability (gray)	
9	Cloud high probability (white)	
10	Thin cirrus (very bright blue)	
11	Snow or ice (very bright pink)	



(a) obtained by a combine harvester (cleaned)

(b) rasterized to Sentinel 2 resolution.

Figure 2.2: Crop yield density map of a field. Ranges from 0.1 t/ha (black) to 5.35 t/ha (white)

136 Note that we call the calculated values merely the *observed NDVI*, as we must be aware
 137 of imprecisions due to clouds and shadows.

138 To define a timescale, we consider Days After Sowing (*DAS*) and a transformed timescale,
 139 Growing Degree Days (*GDD*) ([McMaster and Wilhelm](#) ([McMaster and Wilhelm](#))). The
 140 latter are defined as the cumulative sum (since sowing) of temperature above a given base
 141 temperature T_{base} ⁴. Thus, the GGD for n days after sowing will be equal to:

$$GDD_n := \sum_{i=0}^n \max(T_i - T_{base}, 0). \quad (2.1.3.2)$$

142 Now we create a data set, which will contain all necessary information. Given that we
 143 have the spectral data at a $10m \times 10m$ resolution, we introduce the concept of a Pixel. A
 144 *Pixel P* is associated with a $10m \times 10m$ square defined by the S2 satellites and contains
 145 all relevant information for a season and this location. More precisely, P is a collection
 146 of general information (like yield and coordinates) and all associated P_t of a given season.
 147 Where P_t represents a tuple of the spectral data for time t , the NDVI calculated from it,

⁴XXX For cereals we use $T_{base} = 0$

148 and the associated GDD. We will call the resulting data set *PIXELS* as it is the collection
149 of all Pixels (over all seasons).

150 Finally we split *PIXELS* randomly into a train (80%) and test (20%) set.

151 **Chapter 3**

152 **Interpolation Methods**

153 In this section, we take a closer look at several interpolation methods, which will be used
154 to interpolate and smooth the NDVI time series.

155 First, we give a brief overview in table 3.1.

156 Second, we define the general setting and discuss a general approach to make the interpo-
157 lation more robust (i.e. reduce the impact of outliers).

158 Later, we introduce and discuss each method.

159 Then, we try to extract the main ingredients of each method to forge our own one.

160 Finally, using leave-one-out cross validation, we tune the parameters (where necessary)
161 and get a first idea of the performance of each method.

162 **3.1 Setting**

We are given data in the form of (x_i, Y_i) for $i = 1, \dots, n$. Assume that it can be represented by

$$Y_i = m(x_i) + \varepsilon_i,$$

where ε_i is some noise and $m : \mathbb{R} \rightarrow \mathbb{R}$ being some (parametric or non-parametric) function.
If we assume that $\varepsilon_1, \dots, \varepsilon_n$ i.i.d. with $\mathbb{E}[\varepsilon_i] = 0$ then

$$m(x) = \mathbb{E}[Y | x]$$

163 Different assumptions on m will lead to the following methods:

164 **3.2 XXX DAS vs GDD**

165 equation 2.1.3.2

166 **3.3 Robustify**

167 Now we discuss a general approach of how to robustify an interpolation. The main idea
168 is to give less weight to observations which have high residuals after the initial (or if we
169 reiterate, the last) fit.

Table 3.1: A short summary of the studied interpolation methods. Important assumptions are stated, pros/cons are listed and it is indicated whether the method supports weighted observations (w) and if the resulting interpolation is bounded w.r.t. a fixed interval (b).

	assumptions	pros	cons	w	b
Savitzky-Golay filter	<ul style="list-style-type: none"> - high frequencies are noise (low.pass filter) - equidistant points - local polynomials 	<ul style="list-style-type: none"> - computationally very fast 	<ul style="list-style-type: none"> - cannot deal natively with missing data (need some interpolation) 	no	(yes)
SG + NDVI	<ul style="list-style-type: none"> - upper envelope - vegetation cannot grow faster than some slope 	<ul style="list-style-type: none"> - biological knowledge 	<ul style="list-style-type: none"> - bad “upper envelope” since weights are not used for the estimation itself 	(no)	(yes)
Loess	<ul style="list-style-type: none"> - local polynomial with points closer to the estimated point are more important 	<ul style="list-style-type: none"> - flexible - generalization of SG - weighting function makes intuitive sense 	<ul style="list-style-type: none"> - computationally expensive 	yes	(yes)
Smoothing Splines	<ul style="list-style-type: none"> - 2cd derivative of function is integrable 	<ul style="list-style-type: none"> - intuitive meaning of penalty - general assumptions - flexible shape 	<ul style="list-style-type: none"> - unbounded 	yes	no
B-Splines (Smoothed)	<ul style="list-style-type: none"> - function can be approximated by a linear combination of B-splines basis functions 	<ul style="list-style-type: none"> - general assumption - flexible shape 	<ul style="list-style-type: none"> - unbounded - no intuitive meaning for smoothing 	yes	no
(Gaussian) Kernel Smoothing		<ul style="list-style-type: none"> - simple - general assumptions 	<ul style="list-style-type: none"> - bandwidth: fails if there are big data-gaps 	yes	yes
Double-Logistic	<ul style="list-style-type: none"> - function first increases then decreases - ndvi has a minimal value 	<ul style="list-style-type: none"> - good for evergreen plants (if snow masks ndvi) - upper envelope 	<ul style="list-style-type: none"> - parameterestimation can go seriously wrong - strange behaviour for long data-gaps 	yes	(yes)
Universal Kriging	<ul style="list-style-type: none"> - function is a realization of a stationary gaussian process 	<ul style="list-style-type: none"> - informative parameters - flexible 	<ul style="list-style-type: none"> - regression to the mean - assumptions clearly not met 	yes	(yes)

¹⁷⁰ Even though the procedure is taken from the robust version of the LOESS smoother (cf.
¹⁷¹ section 3.5.4 and [Cleveland \(Cleveland\)](#)), we discuss it now because we will apply it also
¹⁷² to other interpolation methods.

¹⁷³ XXX¹

Before we describe the procedure, we define a function which will determine the weight given to each observation such that observations with large scaled residuals will have less

¹Note that due to using the median for the normalization, we gain a breakdown point of 50% for outliers in y .

weight. That is the bisquare function B:

$$B(x) := \begin{cases} (1 - x^2)^2, & \text{if } |x| < 1 \\ 0, & \text{else} \end{cases}$$

174 Now, we do something similar to what is done in iteratively reweighted least squares. After
 175 an initial interpolation, update the weights of each observation with

$$w_i^{\text{new}} := w_i^{\text{old}} B\left(\frac{|r_i|}{6 \text{mad}(r_1, \dots, r_n)}\right) \quad (3.3.0.1)$$

176 where $r_i = y_i - \hat{y}_i$ denotes the residuals. We can iterate this reweighting and stop after
 177 several steps or when the change of the values is smaller than some tolerance.

178 Examples of such iterative fits are illustrated in the figures 3.4 3.5, 3.6, 3.4 and 3.7.

179 3.3.1 XXX Our Adjustment:

Since we usually observe outliers with negative residuals we decide to divide the negative residuals by two(XXX) before updating the weights. Furthermore, we want to prevent low-weighted observations to corrupt our estimation of scale (the median) and thus we use the weighted median. This can be defined as

$$\text{med}_{\text{weighted}}(r, w) := \arg \min_{\lambda \in \mathbb{R}} \sum_{i=1}^n |r_i w_i - \lambda|$$

180 for $r, w \in \mathbb{R}^n$

181 3.4 Parametric Regression

182 Parametric Curve estimation tries to fit a parametric function (e.g. a Gaussian function
 183 with parameter μ and σ) to a dataset. In the following, we introduce 2 such parametric
 184 approaches.

185 Optimization Issues

186 Since we aim to minimize the residuals sum of squares over 5 (or 6) parameters, we try
 187 to solve a non-convex optimization problem. Thus, the algorithm² either struggles to find
 188 the global minimum or fails to converge. This was fixed by providing for each parameter
 189 reasonable initial values and generous bounds (which match our experience).

190 3.4.1 Double Logistic

191 The Double Logistic smoothing as described in [Beck, Atzberger, Høgda, Johansen, and Skidmore \(Beck et al.\)](#) heavily relies on shape assumptions of the fitted curve (i.e. the
 192 NDVI time series).

193 Assumptions:

- 194 — There is a minimum NDVI level Y_{\min} in the winter (e.g. due to evergreen plants),
 195 which might be masked by snow. This can be estimated beforehand, taking into
 196 several years into account.

²We used the python function `scipy.optimize.curve_fit`

- 198 — The growth cycle can be divided into an increase and a decrease period, where
 199 the time series follows a logistic function. The maximum increase (or decrease) is
 200 observed at t_0 (or t_1) with a slope of d_0 (or d_1).

The equation of the double-logistic fit is given by:

$$Y(t) = Y_{\min} + (Y_{\max} - Y_{\min}) \left(\frac{1}{1 + e^{-d_0(t-t_0)}} + \frac{1}{1 + e^{-d_1(t-t_1)}} - 1 \right)$$

201 Where the five free parameters: Y_{\max} , d_0 , d_1 , t_0 , t_1 are initially estimated by least squares.
 202 Such fit can be seen in figure 3.1.

203 Similar as for the Savitzky-Golay Filter (cf. section 3.5.3) we reestimate (only once) the
 204 parameters by giving less weight to the overestimated observations and more weight to
 205 the underestimated observations³.

Pros	Cons
<ul style="list-style-type: none"> — Incorporates subject specific knowledge in the case of evergreen plants covered in snow. — Optimized parameters have an intuitive meaning. 	<ul style="list-style-type: none"> — Strong shape assumptions on the NDVI curve. — Parameter optimization might go wrong. This can be mitigated to some extent to provide bounds for the parameters — Strange behavior in regions with little observations. (cf. figure 3.1)

206 3.4.2 Fourier Approximation

Similar as in section 3.4.1 we fit a parametric curve to the data by least squares. Here we take the second order Fourier series:

$$\text{NDVI}(t) = \sum_{j=0}^2 a_j \times \cos(j \times \Phi_t) + b_j \times \sin(j \times \Phi_t)$$

207 where $\Phi = 2\pi \times (t - 1)/n$.

Pros	Cons
<ul style="list-style-type: none"> — Assumption of periodicity can be helpful if we are modelling multiyear grow cycles — Flexible curve shape 	<ul style="list-style-type: none"> — Bad behavior in regions with little data (cf. figure 3.1) — Hard to interpret estimated parameters — Parameter estimation can go wrong. Introducing bounds can help.

208 3.5 Non-Parametric Regression

209 In non-parametric curve estimation, we no longer demand our curve to be fully determined
 210 by several parameters, but we allow it to also dependent on the data. That said, we might
 211 still use some tuning-parameters sometimes.

³For the details on the weights we refer to Beck, Atzberger, Høgda, Johansen, and Skidmore (Beck et al.)

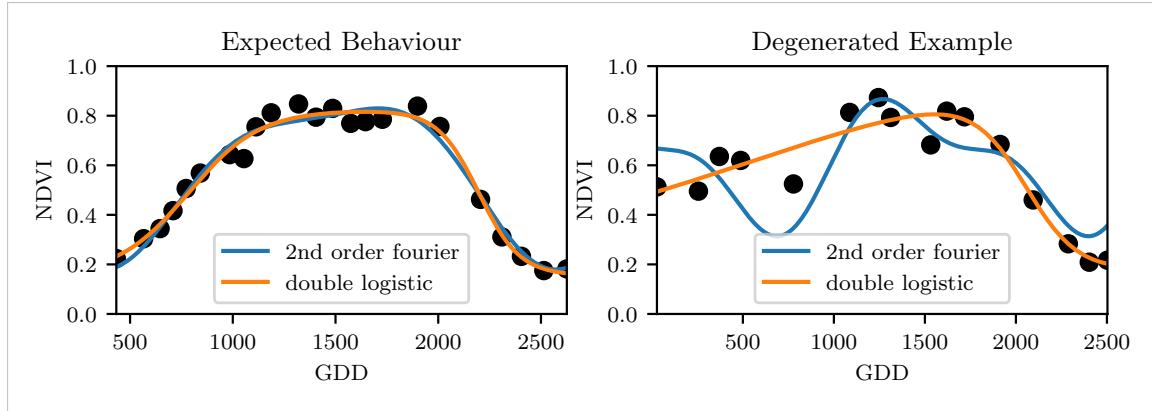


Figure 3.1: Here we observe the nice fitting possibilities of the two parametric methods but notice also some misbehavior

212 3.5.1 Kernel Regression

213 As described previously, we would like to estimate

$$\mathbb{E}[Y | X = x] = \int_{\mathbb{R}} y f_{Y|X}(y | x) dy = \frac{\int_{\mathbb{R}} y f_{X,Y}(x, y) dy}{f_X(x)}, \quad (3.5.1.1)$$

where $f_{Y|X}, f_{X,Y}, f_X$ denote the conditional, joint and marginal densities. This can be done with a kernel K :

$$\hat{f}_X(x) = \frac{\sum_{i=1}^n K(\frac{x-x_i}{h})}{nh}, \quad \hat{f}_{X,Y}(x, y) = \frac{\sum_{i=1}^n K(\frac{x-x_i}{h}) K(\frac{y-Y_i}{h})}{nh^2}$$

By plugging the above into equation 3.5.1.1 we arrive at the *Nadaraya-Watson* kernel estimator:

$$\hat{m}(x) = \frac{\sum_{i=1}^n K((x - x_i)/h) Y_i}{\sum_{i=1}^n K((x - x_i)/h)}$$

Pros	Cons
— flexible due to different possible kernels	— if the $x \mapsto K(x)$ is not continuous, \hat{m} isn't either
— can be assigned degrees of freedom (trace of the hat-matrix)	— choice of bandwidth, especially if x_i are not equidistant.
— estimation of the noise variance $\hat{\sigma}_\varepsilon^2$ (XXX cf. CompStat 3.2.2)	

214 **Examples:** Normal, Box For local bandwidth selection see Brockmann et al. (1993)

215 XXX

216 3.5.2 Kriging

217 Kriging was developed in geostatistics to deal with autocorrelation of the response variable
 218 at nearby points. By applying the notion that two spectral indices which are (timewise)
 219 close should also take similar values, we justify the application of Kriging. In the end, we
 220 would like to fit a smooth Gaussian process to the data. For this subsection, we will follow
 221 Diggle and Ribeiro (dig).

222 **Definitions and Assumptions**

- 223 A *Gaussian Process* $\{S(t) : t \in \mathbb{R}\}$ is a stochastic process if $(S(t_1), \dots, S(t_k))$ has a multi-variate Gaussian distribution for every collection of times t_1, \dots, t_k . S can be fully characterized by the mean $\mu(t) := E[S(t)]$ and its covariance function $\gamma(t, t') = \text{Cov}(S(t), S(t'))$
- 224 Assumption: We will assume the Gaussian process to be stationary. That is for $\mu(t)$ to be
- 225 constant in t and $\gamma(t, t')$ to depend only on $h = t - t'$. Thus, we will write in the following
- 226 only $\gamma(h)$.⁴

We also define the variogram of a Gaussian process as

$$V(h) := V(t, t+h) := \frac{1}{2} \text{Var}(S(t) - S(t+h)) = (\gamma(0))^2(1 - \text{corr}(S(t), S(t+h)))$$

And decide to use a Gaussian Variogram defined by

$$V(h) = p \cdot \left(1 - e^{-\frac{h^2}{(\frac{4}{7}r)^2}}\right) + n,$$

- 229 where h is the distance, n is the nugget, r is the range and p is the partial sill visualized in figure 3.2.⁵

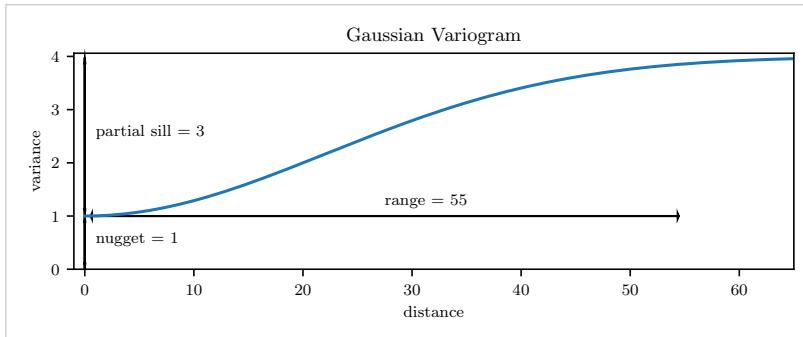


Figure 3.2: Gaussian Variogram with nugget=1, partial sill=3, range=55

230

Pros	Cons
<ul style="list-style-type: none"> — It is a well-studied method. — Parameters have an intuitive meaning. — Flexible covariance structure. 	<ul style="list-style-type: none"> — Regression to the mean. — Violated assumption of constant mean and constant variance. Thus, the NDVI is not a stationary process. — Skewness of errors is not taken into account.

231 **3.5.3 Savitzky-Golay Filter (SG Filter)**

The *Savitzky-Golay Filter*, introduced in [Savitzky and Golay](#) ([Savitzky and Golay](#)) is a technique in signal processing and can be used to filter out high frequencies (low-pass filter) as argued in [Schafer](#) ([Schafer](#)). Furthermore, it also can be used for smoothing by

⁴Note that the process is also *isotropic* (i.e. $\gamma(h) = \gamma(\|h\|)$) since we are in a one-dimensional setting and the covariance is symmetric.

⁵Strictly speaking we use a scaled version of the variogram. Thus, only the ratio of p/n matters.

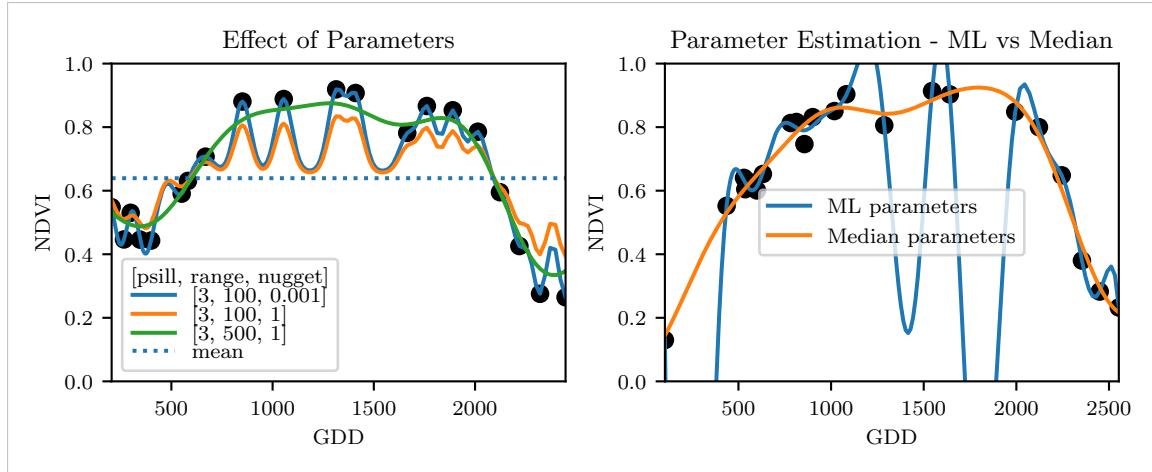


Figure 3.3: On the left, we see how the interpolation change if we increase the nugget and the range parameter. On the right we compare two kriging interpolations, where one takes parameters by numerically maximizing the (which results in a very small nugget) and the other takes the median of many such numerical optimizations.

filtering high frequency noise while keeping the low frequency signal. First, we choose a window size m . Then, for each point, $j \in \{m, m+1, \dots, n-m\}$ we fit a polynomial of degree k by:

$$\hat{y}_j = \min_{p \in P_k} \sum_{i=-m}^m (p(x_{j+i}) - y_{i+j})^2,$$

where P_k denotes the Polynomials of degree k over \mathbb{R} .

For equidistant points this can efficiently be calculated by

$$\hat{y}_j = \sum_{i=-m}^m c_i y_{j+i},$$

where the c_i are only dependent on the m and k and are tabulated in the original paper.

Adaptation to the NDVI

In a rather famous paper [Chen, Jönsson, Tamura, Gu, Matsushita, and Eklundh \(Chen et al.\)](#) a “robust” method based on the Savitzky-Golay has been used. The method is based on the assumption that due to atmospheric effects the observed NDVI tends to be underestimated and that it cannot increase too quickly⁶.

Algorithm:

- i.) Remove points which are labeled as cloudy.
- ii.) Remove points which would indicate an increase greater than 0.4 within 20 days.
- iii.) Linearly interpolate to obtain an equidistant time series X^0 .
- iv.) Apply the Savitzky-Golay Filter to obtain a new time series X^1 .

⁶The latter is argued by the biological impossibility of such fast vegetation changes

244 v.) Update X^1 by applying again a Savitzky-Golay Filter. Repeat this until $w^T |X^1 - X^0|$
 245 stops decreasing, where w is a weight vector with $w_i = \min\left(1, 1 - \frac{X_i^1 - X_i^0}{\max_i \|X_i^1 - X_i^0\|}\right)$.
 246 This reduces the penalty introduced by outliers⁷ and by repeating this step we ap-
 247 proach the “upper NDVI envelope”.

Pros	Cons
— Popular technique in signal processing.	— No natural way of how to estimate points which are not in the data.
— Efficient calculation for equidistant points.	— Not generalizable to other spectral indices.
— Upper envelope matches intuition for the NDVI. Therefore, it is robust against outliers with small values.	— Linear interpolation to account for missing data might be not appropriate.
	— No smooth interpolation between two measurements.

248 **Extension: Spatial-Temporal-Savitzky-Golay Filter**

249 One notable adaptation of the Savitzky-Golay is the presented by Cao, Chen, Shen, Chen,
 250 Zhou, Wang, and Yang (Cao et al.). The key difference is the additional assumption of the
 251 cloud cover being discontinuous and that we can improve by looking at adjacent pixels⁸.
 252 Because we are working with rather high resolution satellite data, and we need the variance
 253 in the predictors, we will waive this extension.

254 **3.5.4 Locally Weighted Regression (LOESS)**

255 Introduced by : Cleveland (Cleveland) implemented here Cappellari, McDermid, Alatalo,
 256 Blitz, Bois, Bournaud, Bureau, Crocker, Davies, Davis, de Zeeuw, Duc, Emsellem, Khoch-
 257 far, Krajnović, Kuntschner, Morganti, Naab, Oosterloo, Sarzi, Scott, Serra, Weijmans,
 258 and Young (Cappellari et al.)

259 The Locally Weighted Regression (LOESS) can be understood as a generalization of the
 260 Savitzky-Golay Filter (cf. sec. 3.5.3).

Given a proportion $\alpha \in (0, 1]$, we estimate each y_i separately by fitting a polynomial of order d by weighted least squares. The weights are (usually) defined by

$$w_i(x_j) = \begin{cases} \left(1 - \left(\frac{x_j}{h_i}\right)^3\right)^3, & \text{for } |x_j| < h_i, \\ 0, & \text{for } |x_j| \geq h_i \end{cases}$$

261 where h_i is the minimal distance such that $\lceil \alpha n \rceil$ observations are in the ball $B_{h_i}(x_i)$.⁹ So
 262 for each y_i we only consider a proportion α of the observations.

⁷Here we call a point i an outlier if $X_i^0 < X_i^1$.

⁸Here, we say that a pixel is adjacent if it is the same pixel but from a different year (keeping the same day of the year) or (if not enough of such temporal-adjacent pixel are found) it is spatially adjacent

⁹If too many weights are set to zero, we might end up considering not enough observations and thus get a singular design-matrix (for the least squares estimation). Therefore, we substitute h_i with $1.01h_i$, so that the observation on the boundary of $B_{h_i}(x_i)$ does not get completely ignored.

263 **How does the Robust LOESS differ from the SG Filter?**

264 The LOESS smoother takes a fraction of points instead of a fixed number and therefore
 265 automatically adapts to the size of the data we wish to interpolate. However, we run
 266 into the danger of considering too little observations, since the estimation breaks down if
 267 $[an] < d + 1$. Furthermore, LOESS gives less weight to points further away. This yields a
 268 "smoother" estimate, since when we slide the window (e.g. for estimating the next value)
 269 an influential point at the border does not suddenly get zero weight from being weighted
 270 equally before. Finally, the LOESS also can be used for non-equidistant data and allows
 271 for arbitrary interpolation.

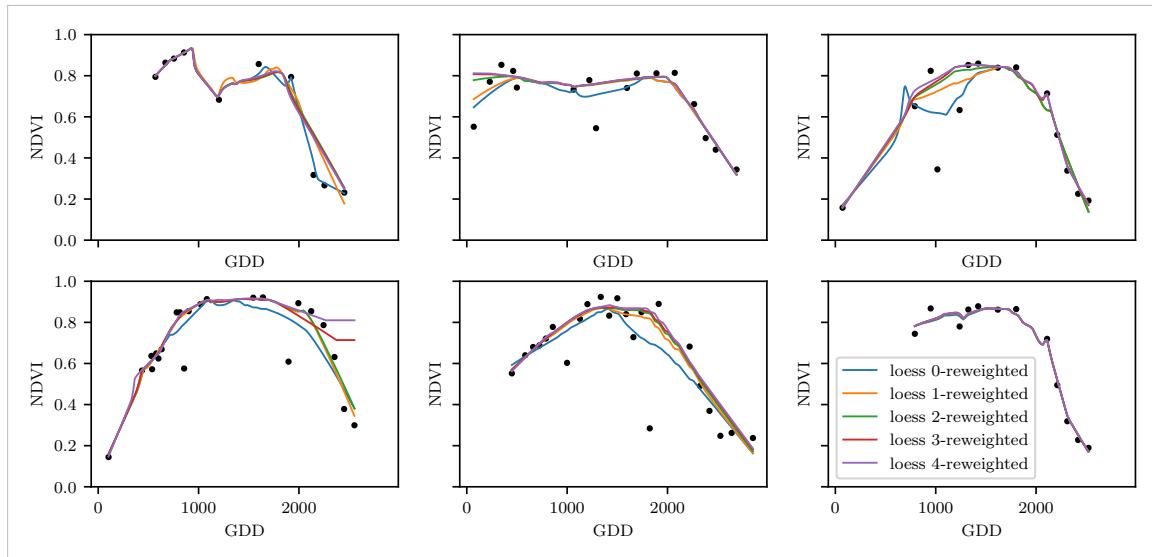


Figure 3.4: The LOESS smoother fitted to different (SCL45) NDVI time series. Iterations of a robustifying refit (as indicated in section 3.3) are also displayed

Pros	Cons
<ul style="list-style-type: none"> — Flexible generalization of Savitzky-Golay — arbitrary interpolation possible — Intuitive parameters 	<ul style="list-style-type: none"> — The nature of local regression might lead to surprising estimates (no smoothness guarantees for the second derivative) — Multiple XXXXXXx

272 **3.5.5 B-splines**

from [Lyche and Mørken](#) ([Lyche and Mørken](#))

$$S(x) = \sum_{j=0}^{n-1} c_j B_{j,k;t}(x)$$

$$B_{i,0}(x) = 1, \text{ if } t_i \leq x < t_{i+1}, \text{ otherwise } 0$$

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

273 ****Smoothing:**** We can relax the constraint that we have to perfectly interpolate. Thus,
 274 we use the minimum number of knots¹⁰ such that: $\sum_{i=1}^n (w(y_i - \hat{y}_i))^2 \leq s$

¹⁰SciPy uses FITPACK and DFITPACK, the documentation suggests that smoothness is achieved by

Pros	Cons
— can be assigned degrees of freedom	— smoothing process does not translate well to a interpretation (unlike smoothing splines)
— extendable to "smooth" version	
— performs also well if points are not equidistant	— choice of smoothing parameter s

275 **3.5.6 Natural Smoothing Splines**

Let \mathcal{F} be the Sobolev space (the space of functions of which the second derivative is integrable). Then the unique¹¹ minimizer

$$\hat{m} := \arg \min_{f \in \mathcal{F}} \sum_{i=1}^n (Y_i - f(x_i))^2 + \lambda \int f''(x)^2 dx$$

276 is a natural¹² cubic spline (i.e. a piecewise cubic polynomial function). The objective
 277 function has an intuitive meaning, as to avoid lateral acceleration it is desirable to move
 278 the steering wheel as little as possible, when driving a car.

Pros	Cons
— can be assigned degrees of freedom (trace of the hat-matrix)	— choose λ
— efficient estimation (closed form solution)	
— intuitive penalty (we don't want the function to be too "wobbly" — change slopes)	
— performs also well if points are not equidistant	
— fixes the Runge's phenomenon (fluctuation of high degree polynomial interpolation)	

279 **3.5.7 XXX Whittaker Smoother**

280 XXX

281 **3.6 Tuning parameter estimation**

282 lots of cross validation

283 what is the best? RMSE is bad, since we know that outliers are present optimizing w.r.t
 284 different statistics

285 ?plot with different densities for each statistic

reducing the number knots used

¹¹Strictly speaking it is only unique for $\lambda > 0$

¹²It is called natural since it is affine outside the data range ($\forall x \notin [x_1, x_n] : \hat{m}''(x) = 0$)

Table 3.2: Performance comparison of different interpolation methods measured with various statistics. Considering only SCL45 points, we get the out-of-bag estimates using the given interpolation method. Consequently, we compute the absolute (value of the) residuals and apply the given statistic to it.

	ss	loess	dl	bspl	fourier	ss rob	loess rob	dl rob	bspl rob	fourier rob
rmse	0.063	0.061	0.061	0.074	0.075	0.070	0.065	0.065	0.079	0.208
qtile50	0.036	0.034	0.027	0.043	0.031	0.032	0.031	0.022	0.037	0.049
qtile75	0.063	0.061	0.051	0.077	0.058	0.061	0.057	0.044	0.070	0.099
qtile85	0.080	0.079	0.070	0.098	0.083	0.081	0.076	0.063	0.094	0.158
qtile90	0.092	0.092	0.088	0.112	0.108	0.097	0.090	0.082	0.113	0.226
qtile95	0.119	0.115	0.122	0.142	0.161	0.132	0.115	0.124	0.157	0.375

286 3.7 Robustification – Recap

- 287 introduced in section ?? we want to review it
 288 robustifieng from loess -> lets try it with all. Result in figures ...
 289 issues when reiterating often (we lose some points completely)
 290 from pictures ... we get that one

291 3.7.1 Upper Envelope Approach - Penalty for negative resiudals

292 discussion of idea, and explenation why we did no use it (arbitrary choice)

293 3.8 Performance Assecement

294 TEMP — Figures

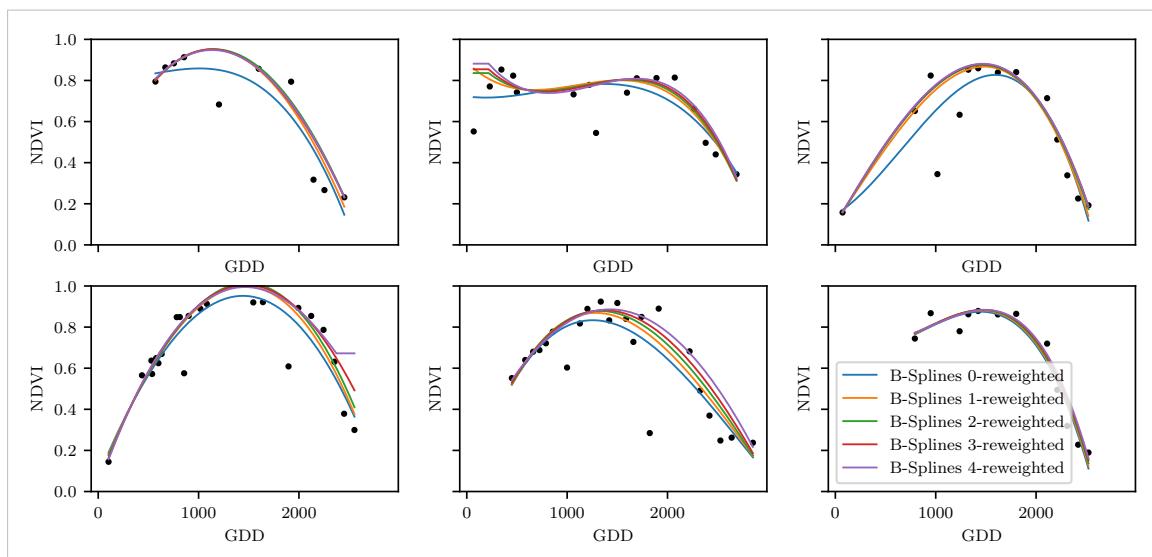


Figure 3.5: B-Splines fitted to different (SCL45) NDVI time series. Iterations of a robustifying refit (as indicated in section 3.3) are also displayed

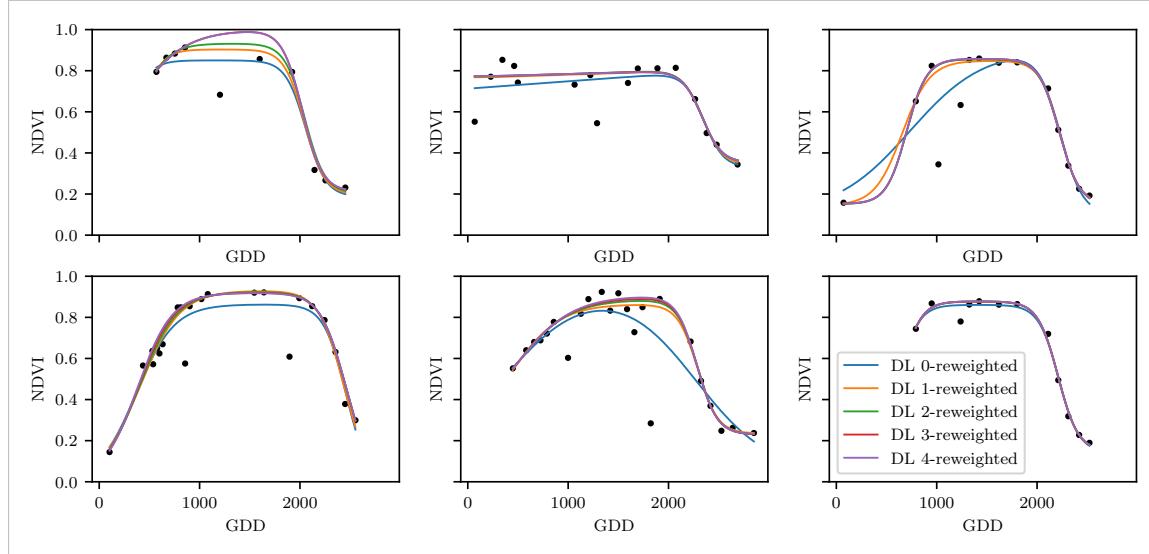


Figure 3.6: A Double Logistic curve fitted to different (SCL45) NDVI time series. Iterations of a robustifying refit (as indicated in section 3.3) are also displayed

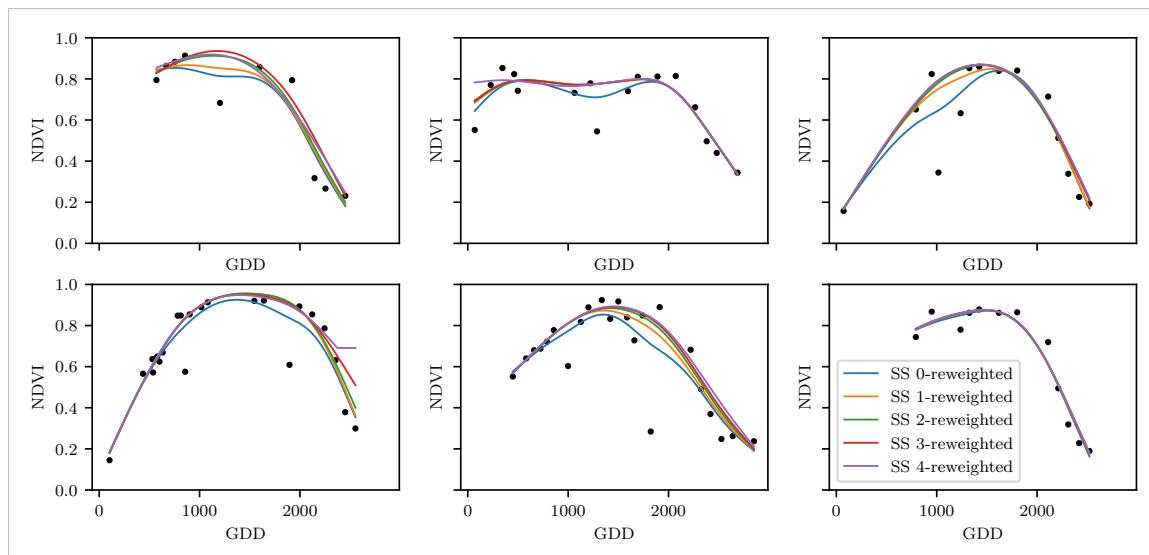


Figure 3.7: Smoothing Splines fitted to different (SCL45) NDVI time series. Iterations of a robustifying refit (as indicated in section 3.3) are also displayed

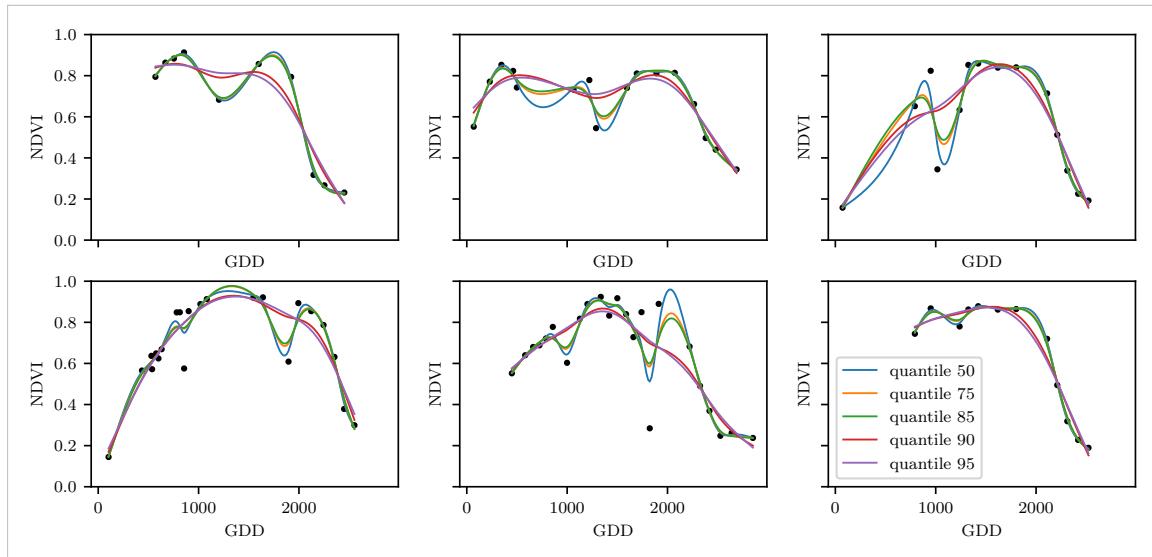


Figure 3.8: Smoothing splines fit with smoothing parameter optimized by minimizing the “...”-quantile of the absolute leave-one-out residuals. Note that the larger the considered quantile is, the smoother the resulting curve becomes.

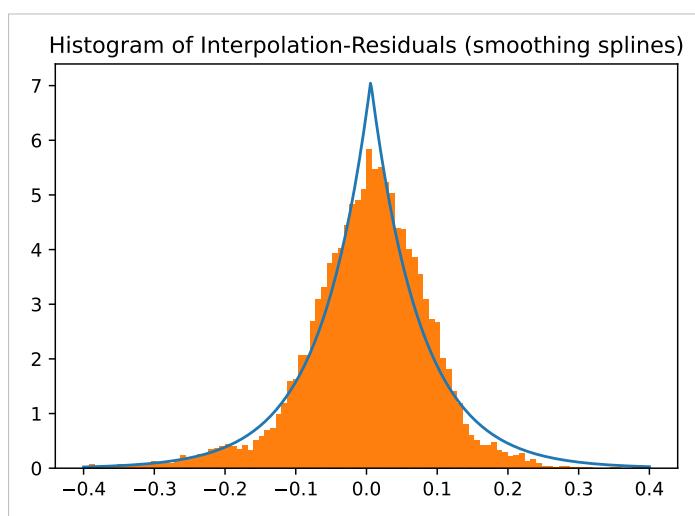


Figure 3.9: XXX caption XXX

295 **Chapter 4**

296 **NDVI Correction / Improve NDVI
297 Data**

298 Let's remind ourselves that the data from the Sentinel-2 is equipped with a scene classi-
299 fication layer (SCL) and we therefore have some information of what is observed at each
300 pixel for each sampled time (cf. table 2.2). In this chapter we would like to improve the
301 observed NDVI values by using more information than just the two bands used to calculate
302 the NDVI (B4 and B8).

303 **4.1 Considering other SCL Classes**

304 In figure 4.1 we see for example that some blue points¹ follow the interpolated line closely
305 and that they might be useful in improving an interpolation fit.

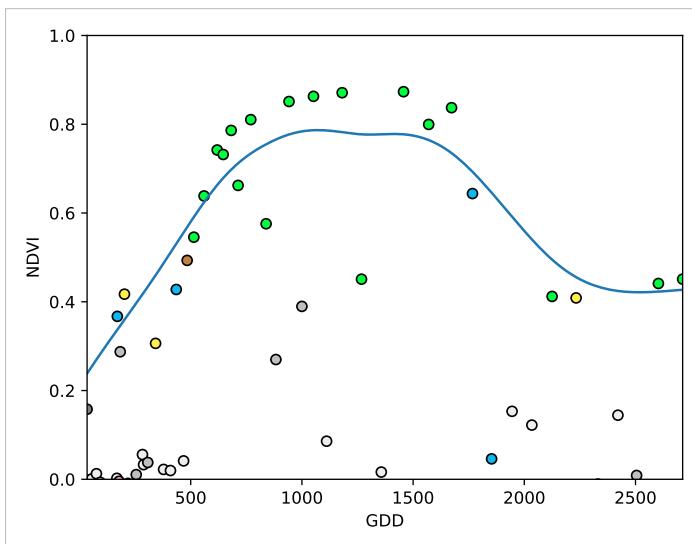


Figure 4.1: A smoothing splines fit considering green and yellow points (SCL45)

306 To get an impression whether there is some useful information contained in the remaining
307 SCL-classes (all except 4 and 5) we would like to compare the observed NDVI with the

¹The blue points correspond to the SCL-class 10: Thin cirrus clouds

308 true NDVI. But since we do not have any ground truth data, we will make the following
 309 assumption:

310 **Definition 4.1.0.1.** *XXXAssumption (true NDVI)* *The true NDVI value at time t can be*
 311 *successfully estimated by out-of-bag interpolation using high quality observations. That is*
 312 *the interpolated value (using XXX) considering the points $P^{SCL45} \setminus P_t$. In the following,*
 313 *we will call this estimate the “true”-NDVI*

314 shall pair every observed NDVI value with its out-of-bag-estimate. Then for each category
 315 we collect all pairs and create a scatter plot in fig 4.2XXXXXXXXXXXXXX

- 316 i.) For each pixel and for each observation (every SCL-class):
 317 estimate the NDVI value (via out-of-the-box interpolation²)
- 318 ii.)

319 4.2 XXX Correction

320 roadmap ... (intuition, data-table, ml-methods, uncertainty, refit and evaluation)

321 4.2.1 XXX idea -and- stepwise plots

322 4.2.2 XXX data-table-construction

323 XXX discussion about choosen covariates: list of things we considered but rejected +
 324 reasoning -> no weather to keep it general even though we have it implemeted

325 4.2.3 XXX ml-methods

326 In the following model we will not explicitly state the intercept³

327 Ordinary Least Squares (OLS)

328 The OLS is a linear model which aims to minimize the sum of the squared residuals. Let
 329 $y \in \mathbb{R}^n$ be the vector of responses and $X \in \mathbb{R}^{n \times p}$ be the design matrix, where each row
 330 corresponds to one pixel and each column consist of one covariate⁴. We assume a linear
 331 relationship between y and X and allow for gaussian noise. That is:

$$y = X\beta + \epsilon \quad \text{where } \epsilon \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2) \quad (4.2.3.1)$$

332 Assuming that X is regular, we can estimte the regression coefficients β by

$$\hat{\beta} = (X^T X)^{-1} X^T y = \arg \min_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2 \quad (4.2.3.2)$$

333 We will train two models, one using only the SCL-classes as covariates and the other one
 334 using all covariates (which are discussed in section 4.2.2).

²That is, we use all observations (in SCL45) but the current one.

³Without loss of generality we can assume that $\bar{y} = 0$ (otherwise consider the centerd version).

⁴Strictly speaking since SCL-classes are dummy variables

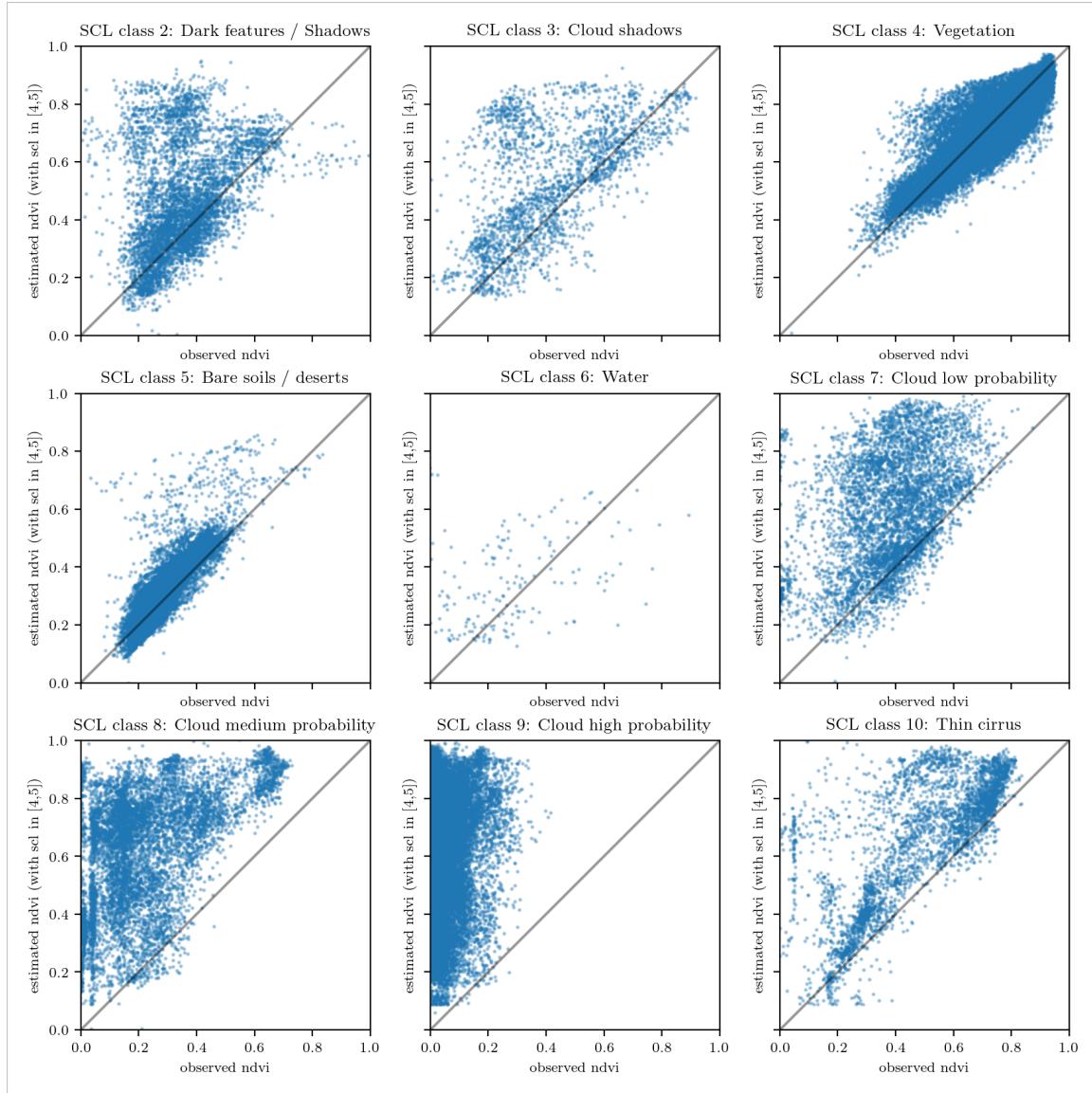


Figure 4.2: XXX caption XXX

335 **LASSO**

336 The Lasso can be similarly expressed than the OLS but adds a penalty to the minimization
 337 problem:

$$\hat{\beta}_\lambda = \arg \min_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 = \arg \min_{\beta \in \mathbb{R}^p \text{ and } \|\beta\|_1 < \lambda} \|y - X\beta\|_2^2. \quad (4.2.3.3)$$

338 Even though we do not have a closed form solution for equation 4.2.3.3 we can solve it
 339 easily via optimization, since the function $\beta \in \{\beta \in \mathbb{R}^p | \|\beta\|_1 < \lambda\} \mapsto \|y - X\beta\|_2^2$ is
 340 continious and convex.

341 Tibshirani (Tibshirani) shows that the LASSO solution tends to be sparse (for not to big
 342 λ). That is $\beta_i = 0$ for most $i = 1, \dots, p$

⁶The last two terms are equivalent by lagrangian optimization

Pros	Cons
<ul style="list-style-type: none"> — Simple method with good interpretability of coefficients. — Computationally cheap. 	<ul style="list-style-type: none"> — Catches only linear relationships. — No integrated variable selection.⁵

343 In order to know which λ to choose we try a huge range of possible values. For each β_λ we
 344 calculate the cross-validated $RMSE_\lambda$ ⁷ (and its standard deviation σ_λ using the k folds)
 345 and define the λ with the smallest corresponding $RMSE_\lambda$ as λ_{min} . From here we choose
 346 the largest λ for which the $RMSE_\lambda$ is smaller than $RMSE_{\lambda_{min}} + \sigma_\lambda$. This yields a simpler
 347 model while keeping the $RMSE$ reasonable model.

348 We will apply the Lasso using the selected covariates in section 4.2.2 and their second
 349 degree of interactions.⁸

Pros	Cons
<ul style="list-style-type: none"> — Usually yields a sparse solution. This tends to give better generalizability (prediction performance on unseen data). — Successfully deals with correlation in covariates. — Interpretable results. 	<ul style="list-style-type: none"> — Estimate is biased. — Computationally expensive.

350 Random Forest

351 To define a random Forest introduced by Breiman (Breiman) we will first define what a
 352 Tree is. A (*decision*) Tree is a graph (V, E) without circles, a distinct root node, every
 353 node has at most two children and every leaf has a value assigned to it. At each node there
 354 is a boolean condition testing if one variable is greater than some value and a pointer to
 355 one child depending on the boolean value. To evaluate a tree we start at the root node,
 356 test the boolean expression and go to the node indicated by the resulting pointer. This
 357 we repeat until we end up at a leaf-node where we return the value assigned to it.

358 To build such a Tree we will recursively partition the covariate space using greedy splits⁹
 359 decreasing the RMSE¹⁰ each time. If the set we want to split contains less than a certain
 360 amount of training points we stop.

361 To build a *Random Forest* we will bootstrap-aggregate¹¹ many such Trees¹². The prediction
 362 of the Random Forest for a new point x is then the mean of the predictions from all

⁷The cross-validate Root Mean Square Error is the mean of the RMSE's obtained for each fold (using the model trained on the remaining folds). We use the following definition of the $RMSE$: $\sqrt{\sum_{i=1}^n (y_i - \hat{y}_i)^2 / n}$

⁸This is if our covariates are $\{a, b\}$, then we will now use $\{a, b, ab, a^2, b^2\}$.

⁹For computational reasons we will only use splits along one covariate. So we 'cut' our covariate space into rectangles.

¹⁰To calculate the RMSE we need a prediction. Let P be the current partition, then the predicted value for some $x \in A \in P$ is the mean of the responses of all the points in A (included in the training data).

¹¹That is we will sample (with replacement) n observations from our original data and fit a Tree to this new sample.

¹²Building the Tree, this time we will not test every covariate at each node (for the RMSE minimization) but a node-specific subsample of the covariates.

the Trees.

Pros	Cons
— Captures non-linear relationships.	— Resulting (prediction) function is non-continuous but locally constant.
— Captures all interactions and performs automatic variable selection.	— Computationally expensive.
— Can deal with missing data.	— No interpretability.

363

364 **Multivariate Adaptive Regression Splines (MARS)**

365 [REFFriedman \(Friedman\)](#)

366 A MARS model can be described by

$$g(x) = \sum_{m=0}^M \beta_m h_m(x), \quad (4.2.3.4)$$

367 where the h_m are simple functions (explained later) and the β_m are estimated via least
368 squares.

369 In the building procedure of a MARS model we first select many of those simple functions
370 and later drop some of them to avoid overfitting.

371 For the construction of those simple functions define \mathcal{B} be the set of pairs of ‘hockystick
372 functions’

$$\mathcal{B} := \left\{ (b_1, b_2) \mid (b_1(x), b_2(x)) = \left((x_j - d)_+, (d - x_j)_+ \right), d = X_{1,j}, \dots, X_{n,j}, j = 1, \dots, p \right\} \quad (4.2.3.5)$$

373 and the set $\mathcal{M} = \{1\}$ of all functions currently in the model. Now, consider \mathcal{C} the set of
374 candidate functions-pairs

$$\mathcal{C} := \{(h(\cdot)b_1(\cdot), h(\cdot)b_2(\cdot)) \mid h \in \mathcal{M}, (b_1, b_2) \in \mathcal{B}\} \quad (4.2.3.6)$$

375 and select the pair (which when added to \mathcal{M} and the coefficients refitted) reduces the
376 RMSE the most. Add the selected pair to \mathcal{M} and repeat until the RMSE reduction
377 becomes insignificant.

378 Finally, to avoid overfitting we prune the set \mathcal{M} by optimizing a generalized cross validation
379 score (GCV).¹³

380 To reduce computational complexity, we follow the recommendation from [REFleaps wrapper](#)
381 ([leaps wrapper](#)) and restrict h in equation 4.2.3.6 to be of degree one (so it is also in
382 a pair of \mathcal{B}). Consequently, \mathcal{C} contains functions with a degree of at most 2.

383 **General Additive Model (GAM)**

384 GAMs as described in [Hastie and Tibshirani \(Hastie and Tibshirani\)](#) are a special case of
385 Projection Pursuit Regression, where only the p directions parallel to the coordinate axes
386 are considered. The result is different to a linear model since the coordinate functions are

¹³This means that we perform an iterative procedure to reduce the number of functions in \mathcal{M} . For every function h in \mathcal{M} we compute the model using $\mathcal{M} \setminus \{h\}$. We discard the function which – when excluding from \mathcal{M} – leads to the best GCV score.

Pros	Cons
<ul style="list-style-type: none"> — Catches non-linear relationships. — Interpretability via functions in \mathcal{M} and their coefficients. — Allows for interactions with variable selection. 	<ul style="list-style-type: none"> — Computationally expensive (can be reduced by restricting the degree of interactions).

387 not restricted to be linear but are assumed to be non-parametric functions. The model
 388 can be written as:

$$g_{add}(x) = \mu + \sum_{i=1}^p g_j(x_j).^{14} \quad (4.2.3.7)$$

389 To estimate the non-parametric functions we can use smoothing splines (ref sec. 3.5.6).
 390 For this let \mathcal{S}_j be the function which takes some $z \in \mathbb{R}^n$ and returns the smoothing splines
 391 fitted to $(X_{:,j}, z)$ where the smoothing parameter is optimized by GCV. Since we cannot fit
 392 all g_j simultaneously we will use a strategy named backfitting. We basically cycle through
 393 the indices $1, \dots, p$ and refit \hat{g}_j each time. The following illustrates the procedure:

- 1) $\hat{g}_1 = \mathcal{S}_1(y - \mu)$
 - 2) $\hat{g}_j = \mathcal{S}_j(y - \mu - \hat{g}_1(X_{:,1}) - \dots - \hat{g}_{j-1}(X_{:,j-1}))$ for $j = 2, \dots, p$
 - 3) $\hat{g}_1 = \mathcal{S}_1(y - \mu - \hat{g}_2(X_{:,2}) - \dots - \hat{g}_p(X_{:,p}))$
 - 4) $\hat{g}_j = \mathcal{S}_j(y - \mu - \sum_{k \neq j} \hat{g}_k(X_{:,k}))$ for $j = 2, \dots, p$
- \vdots

394 We repeat step 3) and 4) until the change falls below some tolerance.

Pros	Cons
<ul style="list-style-type: none"> — Captures non-linearity. — Good interpretability. 	<ul style="list-style-type: none"> — No automatic variable selection. — Computationally expensive.

395 4.2.4 XXX Uncertainty

396 abs(residuals), train models for uncertainty, estimate residuals, get weights (via weight-
 397 function) (problem of weight function -> we should norm the weights somehow since
 398 smoothing parameters are “dependent” on weights -> then, some outer points get really
 399 low weights (just because others in the middle have very little residuals and thus very high
 400 weight))

401 4.3 XXX Evaluation Method

402 yield estimation is a main goal. Claim that yield-estimation-accuracy is a objective mea-
 403 sure : - we have not looked at the yield so far - if the one NDVI-time-series predicts the
 404 yield better than a different one, we conclude that the first time-series carries more true
 405 information about the plants Now: “yield NDVI-TS / derived-covariates”

¹⁴where g_j is a real-valued function. For identifiability we also demand $\mathbb{E}[g_j(X_{:,j})] = 0$ for $j = 1, \dots, p$.

406 **4.3.1 yield estimation**

407 problem: high dimensionality and unequal duration/length -> use features

408 name approaches for yield estimation (we will use a simple but flexible one)

409 random forest ■ for evaluation out-of-bag estimates

410 **Covariates used**

411 reference to kamir et al, why we did choosed some and not others

412 **Chapter 5**

413 **Results**

414 **5.1 XXX small recap from “Interpolation Methods”**

415 **5.2 Robustification and NDVI-Correction**

Table 5.1: XXX RMSE of yield prediction

	rf	lm-scl	lm-all	mars	gam	lasso	no-correction
ss	1.999	1.872	1.829	2.055	2.047	2.033	1.941
dl	1.873	1.886	1.896	1.988	1.898	1.833	2.018
ss-rob	1.895	2.010	2.037	1.970	1.874	1.928	1.880
dl-rob	1.865	1.884	2.002	1.996	1.808	1.875	2.005

416 **Chapter 6**

417 **Discussion**

418 **High RMSE in ...:** How much can we expect to get? We have multiple sources of uncer-
419 tainty in the data: 1. Uncertainty in Yield data collected by the combine harvester 2.
420 Uncertainty in Yield data through rasterization 3. Uncertainty in satellite images through
421 “measurement errors” introduced via clouds and other atmospheric effects 4. Uncertainty
422 introduced by interpolating (especially when long data-gaps are present)

423 **Chapter 7**

424 **Outlook**

425 **7.1 Data**

- 426 — Method how data has been extrapolated to the grid could possibly be improved
427 — For computational reasons we mostly considered all years and split the data (on the
428 pixel level) randomly into a train/test set. A cross Validation with leaving one year
429 out would be

430 **7.2 Interpolation**

- 431 — Penalized Regressions as described in ... are similar to smoothing splines (cf. ...)
432 but different. Better?

433 **7.3 NDVI Correction**

- 434 — try different link functions in section ... between estimated absolute residuals and
435 weights

436 **7.4 NDVI Correction + +**

- 437 — NDVI Correction can be applied to all sorts of land observed via. satellites (without
438 the need of ground truth data)
439 — The idea of NDVI Correction could be applied to other spectral indices like the
440 Green Leaf Area Index.
441 — Yield is not the only target variable of interest. Other variables like protein content
442 could also be used in section ... for the method evaluation.

443 Bibliography

- 444 Gaussian models for geostatistical data. In P. J. Diggle and P. J. Ribeiro (Eds.), *Model-
445 Based Geostatistics*, pp. 46–78. Springer.
- 446 Beck, P. S. A., C. Atzberger, K. A. Høgda, B. Johansen, and A. K. Skidmore. Improved
447 monitoring of vegetation dynamics at very high latitudes: A new method using MODIS
448 NDVI. *100*(3), 321–334.
- 449 Breiman, L. Random Forests. *45*(1), 5–32.
- 450 Cao, R., Y. Chen, M. Shen, J. Chen, J. Zhou, C. Wang, and W. Yang. A simple method to
451 improve the quality of NDVI time-series data by integrating spatiotemporal information
452 with the Savitzky–Golay filter. *217*, 244–257.
- 453 Cappellari, M., R. M. McDermid, K. Alatalo, L. Blitz, M. Bois, F. Bournaud, M. Bu-
454 reau, A. F. Crocker, R. L. Davies, T. A. Davis, P. T. de Zeeuw, P.-A. Duc, E. Em-
455 sellem, S. Khochfar, D. Krajnović, H. Kuntschner, R. Morganti, T. Naab, T. Oosterloo,
456 M. Sarzi, N. Scott, P. Serra, A.-M. Weijmans, and L. M. Young. The ATLAS3D project -
457 XX. Mass-size and mass-sigma distributions of early-type galaxies: Bulge fraction drives
458 kinematics, mass-to-light ratio, molecular gas fraction and stellar initial mass function.
459 *432*, 1862–1893.
- 460 Chen, J., P. Jönsson, M. Tamura, Z. Gu, B. Matsushita, and L. Eklundh. A simple method
461 for reconstructing a high-quality NDVI time-series data set based on the Savitzky–Golay
462 filter. *91*(3), 332–344.
- 463 Cleveland, W. S. Robust Locally Weighted Regression and Smoothing Scatterplots.
464 *74*(368), 829–836.
- 465 Friedman, J. H. Multivariate Adaptive Regression Splines. *19*(1), 1–67.
- 466 Hastie, T. and R. Tibshirani. Generalized Additive Models: Some Applications. *82*(398),
467 371–386.
- 468 Jaramaz, D., V. Perović, S. Belanovic Simic, E. Saljnikov, D. Cakmak, V. Mrvić, and
469 L. Zivotic. The ESA Sentinel-2 mission Vegetation variables for Remote sensing of
470 Plant monitoring.
- 471 leaps wrapper, S. M. D. f. m. b. T. H. a. R. T. U. A. M. F. u. w. T. L. Earth: Multivariate
472 Adaptive Regression Splines.
- 473 Lyche, T. and K. Mørken. Spline Methods.
- 474 McMaster, G. S. and W. W. Wilhelm. Growing degree-days: One equation, two interpre-
475 tations. *87*(4), 291–300.

- 476 Savitzky, A. and M. J. E. Golay. Smoothing and Differentiation of Data by Simplified
477 Least Squares Procedures. *36*(8), 1627–1639.
- 478 Schafer, R. W. What Is a Savitzky-Golay Filter? [Lecture Notes]. *28*(4), 111–117.
- 479 Tibshirani, R. Regression shrinkage and selection via the lasso: A retrospective. *73*(3),
480 273–282.

481 **Appendix A**

482 **XXX Appendix**

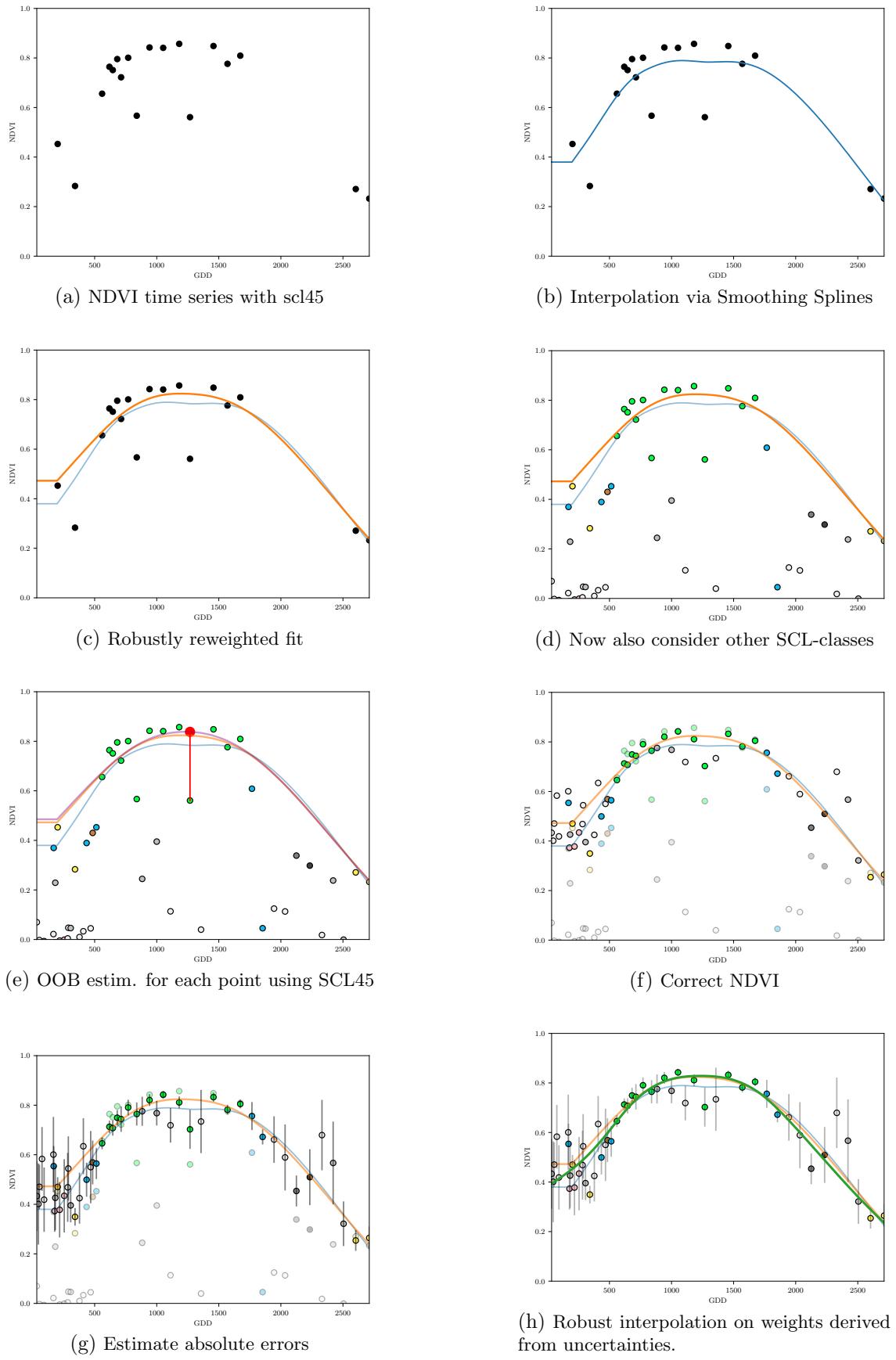


Figure A.1: Stepwise illustration of robust NDVI-Correction