



Swiss Federal Institute of Technology Zurich

Seminar for  
Statistics

1   **Department of Mathematics**

2

3

---

4

5   Master Thesis

Spring 2022

6

---

7

**Lukas Graz**

8

9                   **Interpolation and Correction**

10                  of

11                 **Multispectral Satellite Image Time Series**

11

---

12

Submission Date: September 18th 2022

13

---

14

Co-Adviser: Gregor Perich  
Adviser: Prof. Dr. Nicolai Meinshausen

# 15 Preface

## 16 Supplementary Material

17 GitHub: <https://github.com/LGraz/MasterThesis-Code>

18 R package: <https://github.com/LGraz/CorrectTimeSeries>

## 19 Acknowledgements

20 First, I wish to express my sincere gratitude to my supervisor Prof. Dr. Nicolai Mein-  
21 shausen who took the responsibility for my work and happily took the time to discuss  
22 conceptual and guiding questions and to inspire me with new ideas.

23 It is necessary to highlight that without Gregor Perich this project would not have been  
24 possible. His high personal commitment, reliability as well as the weekly instructive su-  
25 pervision meetings were, without question, essential for this work.

26 It was a real pleasure for me to be part of the *Crop Science* group for this time. Enjoying  
27 everyday company, a two-day excursion, and harvesting wheat together have made this  
28 time truly remarkable. In particular, I would like to thank Prof. Dr. Achim Walter, who  
29 supported this collaboration at its core.

30 Last but not least, I would like to express my gratitude to the *Seminar for Statistics*,  
31 which created the framework conditions for this work and did everything to help me with  
32 conceptional and administrative questions. I should also mention the computing resources  
33 provided by them, without which my computations would not have been feasible.

# 34 Abstract

35 Die Kern-Resultate müssen auch in den Abstract. Ebenso würde ich die vollständige  
Reproduzierbarkeit und die R-Package erwähnen.

- 36 Kurze problemerläuterung (NDVI-ts im Zentrum)
- 37 NDVI Interpolation gewinner
- 38 erforscht Robusification
- 39 NDVI Correction + yield-based evaluation

40 **Contents**

## 41 Todo list

# 42 Notations

## 43 Variables

Hier u.U. nach dem Doppelpunkt ein Absatz/Tab einführen, s.d. die definition und die Erklärung räumlich separiert sind?

44

45  $c$ : a (vector of) constant(s)

46  $\lambda \in \mathbb{R}$ : a scalar

47  $n \in \mathcal{N}$ : sample size

48  $i, j$  are indices in  $\{1, \dots, n\}$

49  $x \in \mathbb{R}^n$ : covariate in 1-dim interpolation setting

50  $w \in \mathbb{R}^n$ : a vector of weights for each location  $x$

51  $y \in \mathbb{R}^n$ : response in 1-dim interpolation setting

52  $\hat{y} \in \mathbb{R}^n$ : estimate of  $y$

53  $\bar{y} \in \mathbb{R}$ : sample mean of  $y$

54  $r \in \mathbb{R}^n$ : residuals given by  $y - \hat{y}$

## 55 Abbreviations and Objects

56 Pixel: A pixel originates of an image pixel and describes a square of 10 x 10 meters in  
57 the field which coincides with the resolution (and location) of the Sentinel-2 pixels. Such  
58 pixels are illustrated in figure ???. Additional information like yield is also attached.

59  $P_t$ : this describes the observed data (weather and spectral bands) at time  $t$  and the location  
60 of one pixel.

61  $P$ : a pixel. We see it as a collection of all the observations at the specified location within  
62 one season. More formally,  $P := \{P_t | t \text{ is a valid sample time within a defined season}\}$

63 SCL: Scene Classification Layer provided by the European Space Agency (ESA) that gives  
64 an estimation of the land cover class of each pixel. It indicates what one can expect at a  
65 pixel at a sampled time. For an overview, c.f. table ??

66  $P^{SCL45}$ : similar to  $P$  but we only consider observations which belong to the classes 4 and  
67 5. This is used done to get a subset of observations which are less contaminated by clouds  
68 and shadows.

- 69 NDVI: Normalized Difference Vegetation Index (?)
- 70 DAS: Days After Sowing
- 71 GDD: growing degree days – cumulative sum of (temperature – threshold)<sup>+</sup>
- 72 XXX ML models and their shortnames
- 73 RYEA : relative yield-estimation-accuracy. Definition ??
- 74 OOB : out-of-box. Describes the procedure if we estimate the value for a point but not
- 75 consider the point itself (cf. section ??)

## 76 MATLAB Matrix Notation

- 77 We will use the MATLAB ‘:’ notation to indicate rows and columns of a matrix. That is
- 78 if  $X \in \mathbb{R}^{n \times p}$  is a matrix, then  $X[:,3]$  is the 3rd column of  $X$  and  $X[2,:]$  is the second row of
- 79  $X$ .

80 **Chapter 1**

81 **Introduction**

82 **1.1 XXX motivation - why is it important**

83 - NDVI-timeseries is simple and widely used. Examples are: - Plant Models REF - Season  
84 Start (start of spring) (community name: land-surface-plant-phenology) - Yield prediction  
85 - crop classification

86 - NDVI is not only of interest to researchers but also public agents and insurance companies

87 Since satellite images are “for free” researchers extract it (only S2 for free)

88 Please also add some words on the S2 satellites of ESA in the introduction.

89 “Similarly, smoothing the time series of satellite data is helpful to address inconsistency  
90 in observation frequency and timing due to clouds and other sensor artefacts ?”

91 **1.2 XXX problembaum / fragestellungen**

92 problemschilderung anhand referenzen und evtl. eines bileds:

93 **1.3 XXX State-of-the-art**

94 Why do we do interpolation in NDVI (and other indices) time series? What are possible shortcomings thereof?

95 zusammenfassung mit literaturrecherche hier (jetzige antowrt auf problemstellung):

- 96 — Doublelogistic (winter-ndvi)  
97 — parametric / non-parametric approaches  
98 — spatio-temporal approaches

99 **1.4 Research Questions**

100 XXX

**101 1.5 Roadmap – anderer name XXX**

**102 This thesis is structured as follows: XXX**

103 **Chapter 2**

104 **Data and Methods**

105 introduce SCL45

106 **2.1 Available Data**

107 Our study region is a farm of over 800ha, which is located in western Switzerland. From  
108 REF-gregor we acquire satellite image data (section ??), yield maps of several cereals from  
109 2017 to 2021 (section ??), and meteorological data (section ??).

110 **2.1.1 Sentinel 2 Data**

111 The European Space Agency (ESA)<sup>1</sup> freely distributes the high-quality images of the two  
112 Sentinel satellites 2 (S2). Together, both satellites have a revisit time of 5 days at the  
113 Equator and 2-3 days at mid-latitudes. However, in our study region, we only receive an  
114 image every 5 days. In order to decrease the effect of atmospheric conditions like reflections  
115 and scattering, bottom-of-atmosphere, radiometric corrected Level-2A data was used<sup>23</sup>.

116 The S2 images contain 12 spectral bands with spatial resolutions up to 10 meters (see  
117 ??). Bands with a lower resolution (20 and 60 meters) were upscaled to 10 meter  
118 resolution using cubic interpolation (REF gregor perich). Additional to the spectral bands,  
119 the ESA also supplies a Scene Classification Layer (*SCL*) where for each location the  
120 observed subject is assigned to an *SCL-class* (cf. table ??). In chapter ?? we will use this  
121 classification to filter out unreliable data points, considering only SCL-classes 4 and 5.

123 **Data Illustration XXXorXXX Challenges in S2 Data**

124 The figure ?? shows a selection of 6 satellite images of a field, which display our challenges.  
125 In February (image a), we see no vegetation but bare soil. At the beginning of May, we  
126 observe a cloudless dark green field. In (c) heavy cloud cover (*SCL* class 9) leads to a  
127 complete loss of plant information in this S2 observation. Figure (d) shows that the *SCL*

Hier noch erwähnen, dass die *SCL* das Resultat eines Algorithmus der ESA ist. Das kann in der Diskussion dann auch wieder aufgenommen werden

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

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

<sup>3</sup>XXXREF gregor perich "Data prior to March 2018 was only 145 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"

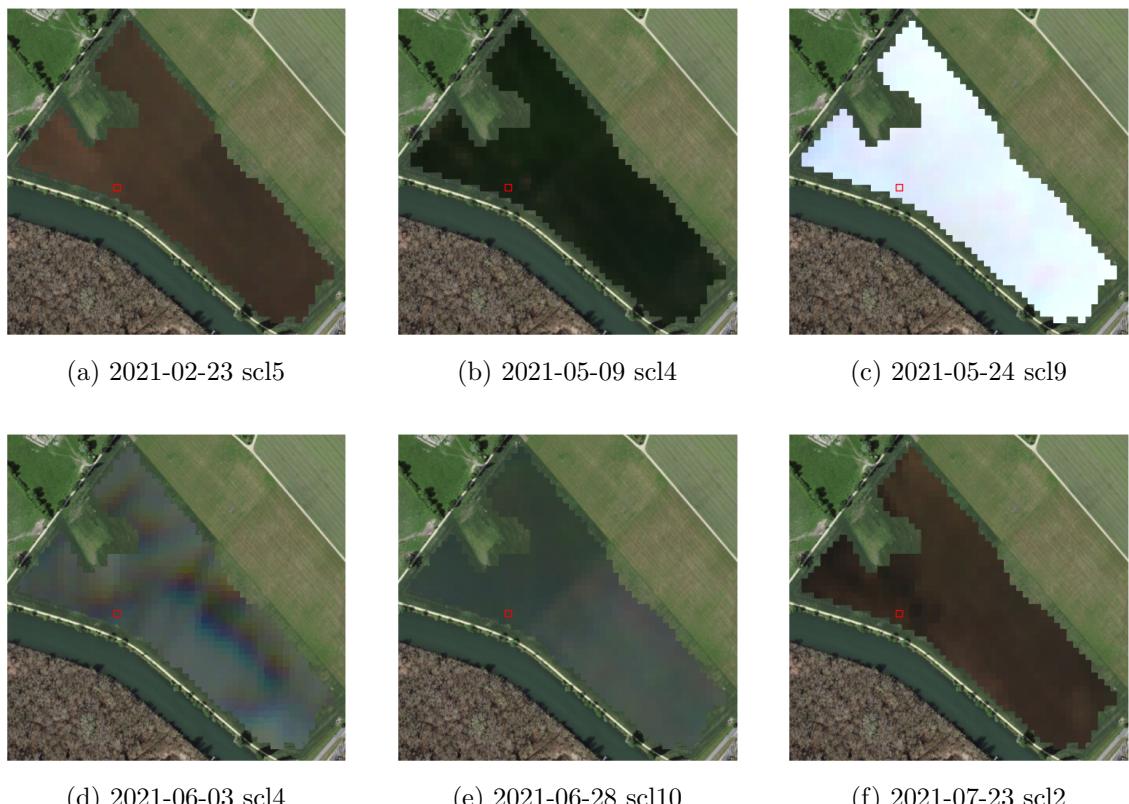


Figure 2.1: Satellite images of a field at selected times with a static background for orientation. The SCL-class of the highlighted pixel is provided in the respective subtitle. (???xxx include scl legend?)

Table 2.1: ? List of spectral bands of the S2-satellites. Each band has its center at the wavelength  $\lambda$  in nm with the spectral width  $\Delta\lambda$  in nm with a spatial resolution  $SR$  in m.

Band	$\lambda$	$\Delta\lambda$	$SR$	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.

128 classification is not reliable, since we evidently observe clouds. In (e) we see a pale green.  
 129 This likely shimmers through cirrus clouds.

### 130 2.1.2 Crop Yield Data

132 The crop yield data were collected using a combine harvester. Equipped with GPS, the  
 133 harvester drives over the fields and continuously estimates the crop density in t/ha (see  
 134 fig. ??). We take the data set derived from this in REF-Gregor-Perich, where error-prone  
 135 measurement points (such as during a tight curve of the combine harvester) were removed  
 136 and then the yield map was rasterized using linear interpolation (cf. fig. ??).

137 Comparing the average per-field crop yield reported by the farmer with the yield estimated  
 138 by the combine harvester shows that the latter overestimates crop yield by ca. 10% (cf.  
 139 REF-gregor). Since the relative estimation error is approximately constant and we do not  
 140 aim to estimate the absolute yield, we will not consider this deviation.

Hier  
bitte  
noch  
eine  
kleine  
Beschrei-  
bung  
der  
Crop-  
yields  
rein-  
nehmen.  
Also  
was sind  
die Er-  
tragswerte  
der  
Daten.

### 141 2.1.3 The Concept of a ‘Pixel’

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

143 The well-known Normalized Difference Vegetation Index (*NDVI*) introduced in ? can be  
 144 calculated using the bands *B4* and *B8* (table ??) by:

$$NDVI = \frac{B8 - B4}{B8 + B4}$$

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

Please  
clarify  
this in  
more  
detail.  
We used  
pixels

Table 2.2: Overview: Scene Classification Layers (SCL)

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

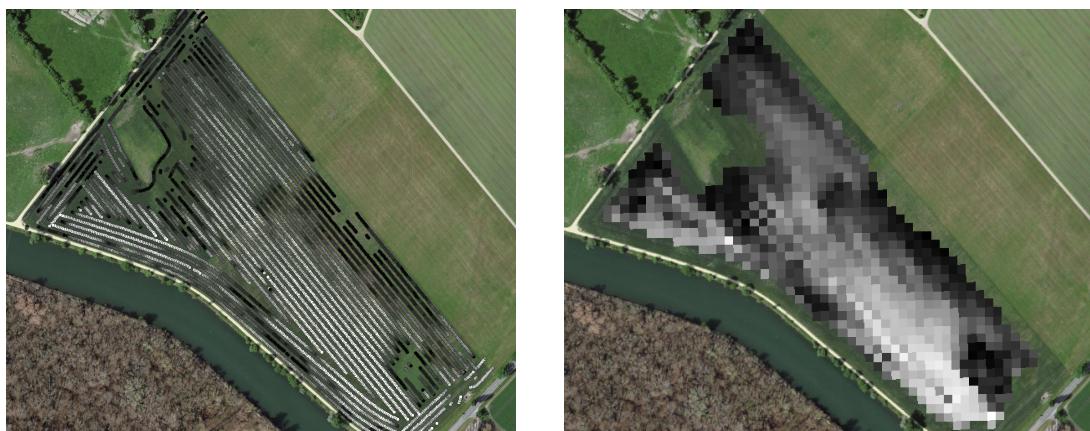


Figure 2.2: Crop yield density map of a field. Ranges from 0.1 t/ha (black) to 5.35 t/ha (white)

To define a timescale, we consider Days After Sowing (*DAS*) and a transformed timescale, Growing Degree Days (*GDD*) (?REF). The latter are defined as the cumulative sum (since sowing) of temperature above a given base temperature  $T_{base}$ . For cereals, we use  $T_{base} = 0$  (REF-Gregor). Thus, the GGD for  $n$  days after sowing will be equal to:

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

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

Für den Leser wäre es interessant, wenn Du noch kurz die wichtigsten GDD Werte aus der Literatur beschreiben würdest (D.h. z.B. Sowing, —

159 **2.2 General Methods**

160 We will only introduce general methods within this section, whereas more specific methods  
 161 will be introduced in their context. We discuss interpolation methods in sections ?? and  
 162 ??, a robustification strategy in section ??, a method how we can objectively determine the  
 163 quality of an interpolation in section ??, and in section ?? we present the NDVI correction  
 164 with an adapted interpolation strategy.

which  
context?  
I would  
write:  
"... in-  
tro-  
duced  
in chap-  
ter/-  
section  
XXX  
in more  
detail."

165 **2.2.1 (Relative) RMSE**

166 definition here and also relative

167 **2.2.2 Out-Of-Bag (*OOB*) and Leave-One-Out-Cross-Validation (*LOOCV*)**

168 Hier fehlt mir eine kurze Erklärung, was OOB und LOOCV sind.

Let

$$D = \{(X_{[j,:]}, y_j) \mid X \in \mathbb{R}^{n \times p}, y \in \mathbb{R}^n, j = 1, \dots, n\}$$

169 be a dataset,  $i \in \{1, \dots, n\}$  and  $M^{(-i)}$  a model fitted on a subset of  $D \setminus \{(X_{[i,:]}, y_i)\}$ . Then  
 170 we call  $\hat{y}_i := M^{(-i)}(X_{[i,:]})$  an *OOB* estimator of  $y_i$ . If we do this for all  $i \in \{1, \dots, n\}$ , we  
 171 obtain  $\hat{y} := (\hat{y}_1, \dots, \hat{y}_n)$  the *OOB* estimator for  $y \in \mathbb{R}^n$ .

172 In the bootstrap (e.g., random forest) framework, we define  $\hat{y}_i$  to be the average of all  
 173 computed and admissible  $M^{(-i)}$ .

174 In the case that  $M^{(-i)}$  was fitted on the set  $D \setminus \{(X_i, y_i)\}$  (i.e., not a true subset), we call  
 175 the corresponding  $\hat{y}_i$  also the *LOOCV* estimator.

176 If we optimize some parameter via *OOB* (or *LOOCV*) this means that we search for the  
 177 parameter that minimizes some loss function which takes the *OOB* (or *LOOCV*) residuals.  
 178 Usually we approximate this parameter by searching on a grid.

179 **2.2.3 Generalized Cross Validation (GCV)**

180 definition here and explanation why (computational cheap)

181 **Chapter 3**

182 **Interpolation Methods**

183

184 In this section, we take a closer look at several interpolation methods, which will be  
185 used to interpolate and smooth the NDVI time series, while considering only SCL45 in  
186 this chapter. A brief overview of the considered interpolation methods can be found in  
187 table ??.

188 First, we define the general setting and discuss a general approach to make the interpola-  
189 tion more robust (i.e. reduce the impact of outliers).

190 Afterwards, we introduce and discuss each method.

191 Then, we try to extract the main ingredients of each method to construct a new one with  
192 all benefits.

193 Finally, using LOOCV, we tune the parameters (where necessary) and get a first idea of  
194 the performance of each method.

verdeutliche  
dem  
leser,  
dass ein  
auftrag  
das  
findne  
von  
interpo-  
lation-  
metho-  
den war

195 **3.1 DAS vs. GDD**

196 Prior to interpolating the NDVI time series, we should decide on a timescale. We can  
197 choose between DAS and GDD (cf. section ?? and equation ??). In figure ?? we see an  
198 example for comparison of the two. Here we see that the first 120 DAS are compressed  
199 to just 500 GDD. This has several advantages. First, it makes the scales comparable (in  
200 terms of plant growth) because the plants are not concerned with the month of the year  
201 but the current temperature. Second, in winter we tend to have higher cloud cover and  
202 thus fewer SCL45 observations. Hence, this gap in observations is compressed. Therefore,  
203 we will only use GDD in the subsequent.

204 **3.2 Interpolation Setup**

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  $\varepsilon_i$  is some noise and  $m : \mathbb{R} \rightarrow \mathbb{R}$  is 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

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

put sec-  
tion in  
methods  
/ data

Findet  
man  
hier  
noch  
Liter-  
atur, in  
welcher  
ähn-  
liches  
disku-  
tiert  
wurde,  
die man  
zitieren  
kann?

add fourier and fix order to match chapters

Table 3.1: Summary of the studied interpolation methods containing important assumptions, advantages and disadvantages and whether the method supports weighted observations (w) and if the resulting interpolation is bounded w.r.t. a fixed interval (b).

	<b>Assumptions</b>	<b>Advantages</b>	<b>Disadvantages</b>	<b>w</b>	<b>b</b>
Savitzky-Golay filter	<ul style="list-style-type: none"> <li>– High frequencies are noise (Low-Pass-Filter)</li> <li>– Equidistant points</li> <li>– Local polynomials</li> </ul>	<ul style="list-style-type: none"> <li>– Computationally very fast</li> </ul>	<ul style="list-style-type: none"> <li>– Cannot deal natively with missing data (need some interpolation)</li> </ul>	No	(Yes)
SG + NDVI	<ul style="list-style-type: none"> <li>– Upper envelope</li> <li>– Vegetation cannot grow faster than some slope</li> </ul>	<ul style="list-style-type: none"> <li>– Biological edge</li> </ul>	<ul style="list-style-type: none"> <li>– Bad “upper envelope” since weights are not used for the estimation itself</li> </ul>	(No)	(Yes)
LOESS	<ul style="list-style-type: none"> <li>– Local polynomial with points closer to the estimated point are more important</li> </ul>	<ul style="list-style-type: none"> <li>– Flexible</li> <li>– Generalization of SG</li> <li>– Weighting function makes intuitive sense</li> </ul>	<ul style="list-style-type: none"> <li>– Computationally expensive</li> </ul>	Yes	(Yes)
Smoothing Splines	<ul style="list-style-type: none"> <li>– 2cd derivative of function is integrable</li> </ul>	<ul style="list-style-type: none"> <li>– Intuitive meaning of penalty</li> <li>– General assumptions</li> <li>– Flexible shape</li> </ul>	<ul style="list-style-type: none"> <li>– Unbounded</li> </ul>	Yes	No
B-Splines (Smoothed)	<ul style="list-style-type: none"> <li>– Function can be approximated by a linear combination of B-splines basis functions</li> </ul>	<ul style="list-style-type: none"> <li>– General assumption</li> <li>– Flexible shape</li> </ul>	<ul style="list-style-type: none"> <li>– Unbounded</li> <li>– No intuitive meaning for smoothing</li> </ul>	Yes	No
(Gaussian) Kernel Smooth-ing	<ul style="list-style-type: none"> <li>– Close points are related to each other via a kernel function</li> </ul>	<ul style="list-style-type: none"> <li>– Simple</li> <li>– General assumptions</li> </ul>	<ul style="list-style-type: none"> <li>– Bandwidth: fails if there are big data-gaps</li> </ul>	Yes	Yes
Double-Logistic	<ul style="list-style-type: none"> <li>– Function first increases then decreases</li> <li>– Ndvi has a minimal value</li> </ul>	<ul style="list-style-type: none"> <li>– Good for evergreen plants (if snow masks NDVI)</li> <li>– Upper envelope</li> </ul>	<ul style="list-style-type: none"> <li>– Parameter estimation can go seriously wrong</li> <li>– Strange behavior for long data-gaps</li> </ul>	Yes	(Yes)
Universal Kriging	<ul style="list-style-type: none"> <li>– Function is a realization of a stationary Gaussian process</li> </ul>	<ul style="list-style-type: none"> <li>– Informative parameters</li> <li>– Flexible</li> </ul>	<ul style="list-style-type: none"> <li>– Regression to the mean</li> <li>– Assumptions clearly not met</li> </ul>	Yes	(Yes)

<sup>206</sup> We will introduce parametric and non-parametric approaches to estimate  $m$  in section ?? and ??.

<sup>208</sup> Furthermore, in the subsequent, we denote  $w \in \mathbb{R}^n$  as the vector of weights such that  $w_i$  corresponds to the weight that  $(x_i, Y_i)$  should have in the interpolation.

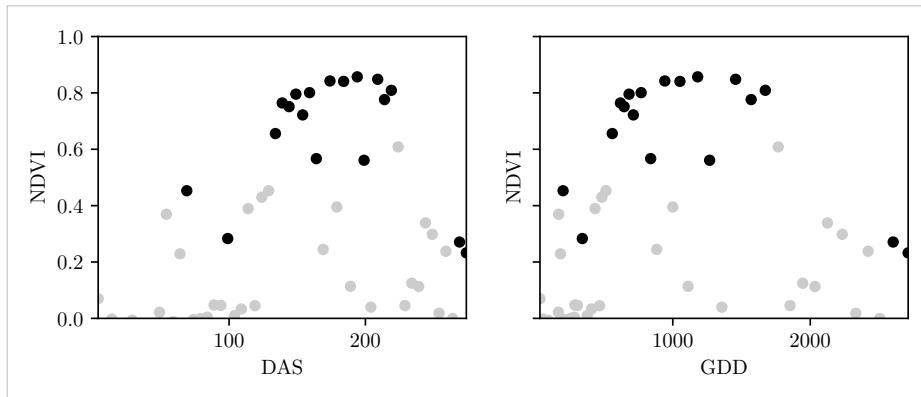


Figure 3.1: The same NDVI time-series, on the left with DAS as the timescale, on the right GDD is the timescale. SCL45 are colored black. Non-SCL45 (clouds and shadows) are colored in gray.

210 Paper zitieren wo eingeführt oder wo benutzt (falls einführung fast schon trivial)

211 Ähnliche struktur sich überlegen

### 212 3.3 Parametric Regression

213 Parametric Curve estimation tries to fit a parametric function, such as, for example, a  
214 Gaussian function with parameters  $\mu$  and  $\sigma$ , to a dataset. In the following, we introduce  
215 two parametric approaches.

#### 216 3.3.1 Double Logistic

217 The Double Logistic smoothing as described in ?REF heavily relies on shape assumptions  
218 of the fitted curve (i.e. the NDVI time series).

219 Assumptions:

220 Die Aufzählung ist hier m.M.n. nicht so passend für einen "Fließtext".

- 221 — There is a minimum NDVI level  $y_{\min}$  in the winter (e.g. due to evergreen plants),  
222 which might be masked by snow. This can be estimated beforehand, taking several  
223 years into account.
- 224 — The growth cycle can be divided into an increase and a decrease period, where  
225 the time series follows a logistic function. The maximum increase (or decrease) is  
226 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:

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

227 Where the five free parameters:  $y_{\max}$ ,  $d_0$ ,  $d_1$ ,  $t_0$ ,  $t_1$  are initially estimated by least squares.  
228 Such fit can be seen in figure ??.

229 Similar as for the Savitzky-Golay Filter (cf. section ??) we reestimate (only once) the  
230 parameters by giving less weight to the overestimated observations and more weight to  
231 the underestimated observations<sup>1</sup>.

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

232 **3.3.2 Fourier Approximation**

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

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

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

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

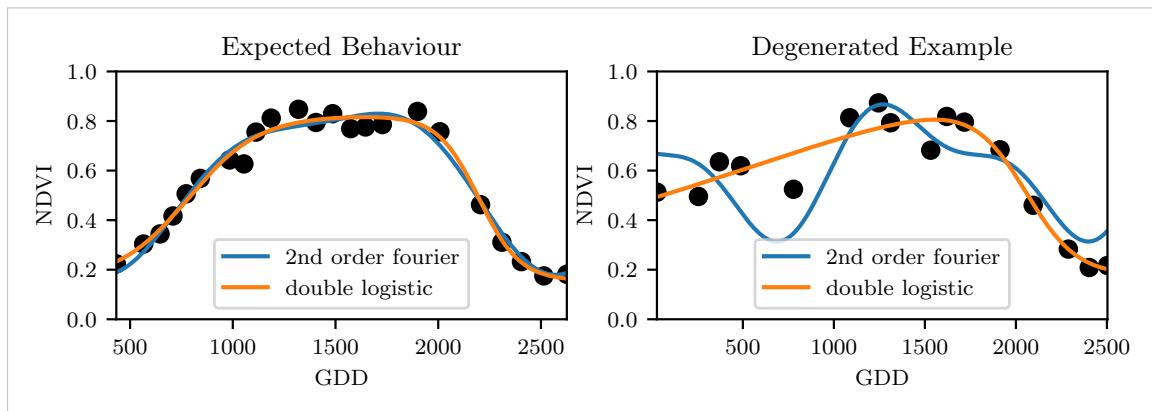


Figure 3.2: Here we observe the possibilities of a precise fit for the two parametric methods but notice also some misbehavior

<sup>1</sup>For the details on the weights we refer to ?

234 **3.3.3 Optimization Issues**

235 We shall mention some optimization issues we countered during implementation. Since we  
 236 aim to minimize the residual sum of squares over 5 (or 6) parameters, we try to solve a  
 237 non-convex optimization problem. Thus, the algorithm<sup>2</sup> either struggles to find the global  
 238 minimum or fails to converge. This was fixed by providing for each parameter reasonable  
 239 initial values and generous bounds (which match our experience).

240 **3.4 Non-Parametric Regression**

242 In non-parametric curve estimation, the curve does no longer have to be fully determined  
 243 by parameters, but we allow it to also depend on the data. Note, that we do not exclude  
 244 the use of tuning-parameters.

TODO:  
include  
Weighted  
versions

245 **3.4.1 Kernel Regression**

246 As described previously (XXX REF Setup section), we would like to estimate

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

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

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

By using the above function in equation (??) we arrive at the *Nadaraya-Watson* kernel estimator:

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

247 Common choices for the kernel are the normal function or a uniform function (also called  
 248 ‘box’ function.). Note that we still need to choose the bandwidth of the function (in  
 249 the case of the normal function, this is  $\sigma$  the standard deviation). For local adaptive  
 250 bandwidth selection we refer to ?.

Advantages	Disadvantages
— 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 $\hat{\sigma}_\varepsilon^2$ (REF cf. CompStat 3.2.2)	

Die  
Aufzäh-  
lung  
ist hier  
m.M.n.  
nicht so  
passend  
für  
einen  
”Fliess-  
text”.

251 **3.4.2 Kriging**

252 Kriging was developed in geostatistics to deal with autocorrelation of the response variable  
 253 at locations which are spatially close. By applying the notion that two spectral indices  
 254 which are (timewise) close should also take similar values, we justify the application of  
 255 Kriging. In the end, we would like to fit a smooth Gaussian process to the data. For this  
 256 subsection, we will follow ?.

<sup>2</sup>We used the python function `scipy.optimize.curve_fit`

ohne  
un-  
terkapi-  
tel  
struk-  
tur

## 258 Definitions and Assumptions

259 **Definition 3.4.2.1.** (*Gaussian Process*) A Gaussian Process  $\{S(t) : t \in \mathbb{R}\}$  is a stochastic  
 260 process if  $(S(t_1), \dots, S(t_k))$  has a multivariate Gaussian distribution for every collection of  
 261 times  $t_1, \dots, t_k$ .  $S$  can be fully characterized by the mean  $\mu(t) := E[S(t)]$  and its covariance  
 262 function  $\gamma(t, t') = \text{Cov}(S(t), S(t'))$

263 **Assumption 1.** We will assume the Gaussian process to be stationary. That is for  $\mu(t)$   
 264 to be constant in  $t$  and  $\gamma(t, t')$  to depend only on  $h = t - t'$ . Thus, we will write in the  
 265 following only  $\gamma(h)$ .<sup>3</sup>

**Definition 3.4.2.2.** (*Variogram*) 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

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

266 where  $h$  is the distance,  $n$  is the nugget,  $r$  is the range and  $p$  is the partial sill visualized  
 267 in figure ??.<sup>4</sup>

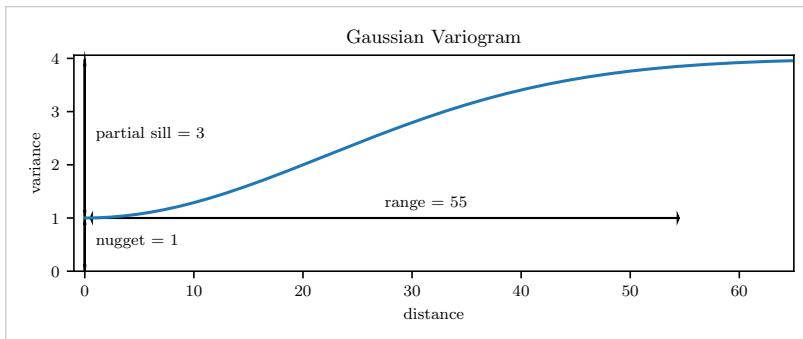


Figure 3.3: Gaussian Variogram with nugget=1, partial sill=3, range=55

268 Next, we consider a one-dimensional Gaussian process  $G_\gamma$  with variogram  $\gamma$ . We tune the  
 269 variogram parameters using maximum likelihood<sup>5</sup>. Let  $z$  be a vector with the new values  
 270 to extrapolate, then we can determine the values  $m(z) = \mathbb{E}[G_\gamma(z)|(x, y)]$  using Bayes  
 271 rule<sup>6</sup>. For an example fit, we refer to figure ??.

272 Since we observe a clear pattern of a growth period in spring and harvest in the end of  
 273 summer, we have to admit that assumption ?? with the constant mean is clearly violated.  
 274 This is also the reason why we observe (for every variogram parameter) a tendency to the  
 275 mean, as indicated in figure ??.

<sup>3</sup>Note 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.

<sup>4</sup>Strictly speaking we use a scaled version of the variogram. Thus, only the ratio of  $p/n$  matters.

<sup>5</sup>As illustrated in figure ?? maximum likelihood estimation can lead to overfitting. Thus, we will in practice sample several such optimized parameters and use their median in the end.

<sup>6</sup>Bayes rule generally claims, that for two random variables  $A$  and  $B$  we have that  $P(A|B) = P(B|A)/P(B)$

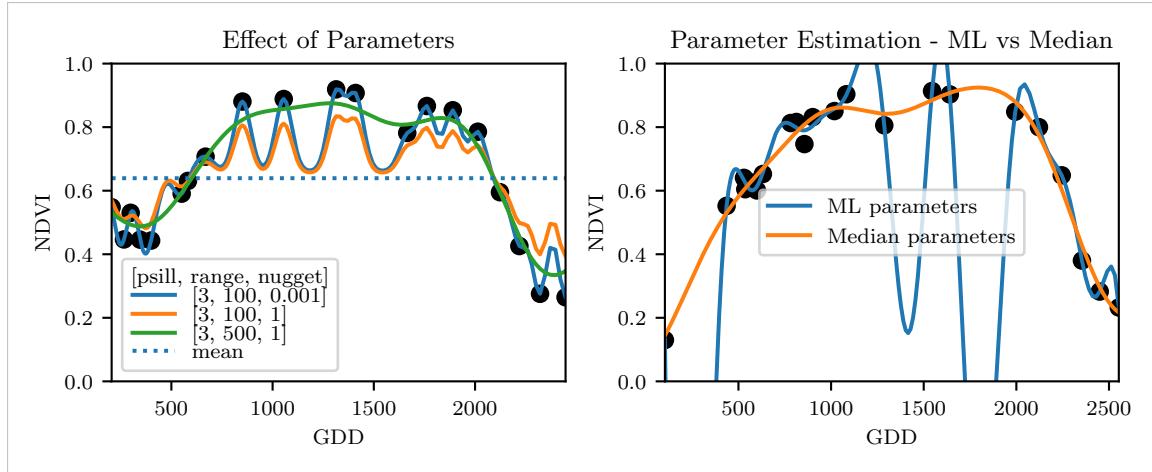


Figure 3.4: On the left, we see how the interpolation change if we increase the nugget and the range parameter. On the right, we compare two kriging interpolations, where one takes parameters by numerically maximizing the (which results in a very small nugget) and the other takes the median of many such numerical optimizations.

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

### 276 3.4.3 Savitzky-Golay Filter (SG Filter)

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

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

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

For equidistant points this can efficiently be calculated by

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

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

### 279 Adaptation to the NDVI

280 ? developed a ‘robust’ interpolation method for the NDVI based on the SG Filter. The  
281 method is based on the assumption that due to atmospheric effects the observed NDVI

figure /  
tabelle  
/ pseudocode  
anstatt  
aufzählung

- 282 tends to be underestimated and that it cannot increase too quickly. The latter is argued  
 283 by the biological impossibility of such fast vegetation changes. Their proposed algorithm  
 284 is:
- 285 i.) Remove points which are labeled as cloudy.  
 286 ii.) Remove points which would indicate an increase greater than 0.4 within 20 days.  
 287 iii.) Linearly interpolate to obtain an equidistant time series  $X^0$ .  
 288 iv.) Apply the SG Filter to obtain a new time series  $X^1$ .  
 289 v.) Update  $X^1$  by applying again a SG Filter. Repeat this until  $w^T |X^1 - X^0|$  stops  
 290 decreasing, where  $w$  is a weight vector with  $w_i = \min\left(1, 1 - \frac{X_i^1 - X_i^0}{\max_i \|X_i^1 - X_i^0\|}\right)$ . This  
 291 reduces the penalty introduced by outliers<sup>7</sup> and by repeating this step we approach  
 292 the “upper NDVI envelope”.

Advantages	Disadvantages
<ul style="list-style-type: none"> <li>— Popular technique in signal processing.</li> <li>— Efficient calculation for equidistant points.</li> <li>— Upper envelope matches intuition for the NDVI. Therefore, it is robust against outliers with small values.</li> </ul>	<ul style="list-style-type: none"> <li>— No natural way of how to estimate points which are not in the data.</li> <li>— Not generalizable to other spectral indices.</li> <li>— Linear interpolation to account for missing data might be not appropriate.</li> <li>— No smooth interpolation between two measurements.</li> </ul>

### 293 Extension: Spatial-Temporal-Savitzky-Golay Filter

294 One notable adaptation of the SG Filter is the presented by ?. The key difference is the  
 295 additional assumption of the cloud cover being discontinuous and that we can improve by  
 296 looking at adjacent pixels<sup>8</sup>. Because we are working with rather high resolution satellite  
 297 data, and we need the variance in the predictors, we will waive this extension.

#### 298 3.4.4 Locally Weighted Regression (LOESS)

299 The Locally Weighted Regression (LOESS) introduced by ? can be understood as a  
 300 generalization of the SG Filter (cf. sec. ??).

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

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

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

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

<sup>8</sup>Here, 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

<sup>9</sup>If too many weights are set to zero, we might end up considering not enough observations and thus

303 **Differences between the Robust LOESS and the SG Filter?**

304 The LOESS smoother takes a fraction of points instead of a fixed number and therefore  
 305 automatically adapts to the size of the data we wish to interpolate. However, we run  
 306 into the danger of considering too little observations, since the estimation breaks down if  
 307  $\lceil \alpha n \rceil < d + 1$ .<sup>??</sup> Furthermore, LOESS gives less weight to points further away. This yields  
 308 a "smoother" estimate, since when we slide the window (e.g. for estimating the next value)  
 309 an influential point at the border does not suddenly get zero weight from being weighted  
 310 equally before. Finally, the LOESS also can be used for non-equidistant data and allows  
 311 for arbitrary interpolation.

Advantages	Disadvantages
<ul style="list-style-type: none"> <li>— Flexible generalization of SG Filter</li> <li>— arbitrary interpolation possible</li> <li>— Intuitive parameters</li> </ul>	<ul style="list-style-type: none"> <li>— The nature of local regression might lead to surprising estimates (no smoothness guarantees for the second derivative)</li> </ul>

312 **3.4.5 B-splines**

B-splines as discussed in ? are piecewise cubic polynomials defined by

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

where  $B$  are basis functions and recursively defined by:

$$\begin{aligned} B_{i,0}(z) &= 1, \text{ if } t_i \leq z < t_{i+1}, \text{ otherwise } 0 \\ B_{i,k}(z) &= \frac{z-x_i}{x_{i+k}-x_i} B_{i,k-1}(z) + \frac{x_{i+k+1}-z}{x_{i+k+1}-x_{i+1}} B_{i+1,k-1}(z). \end{aligned}$$

Assuming that all  $x_i$  are distinct, this yields an interpolation which fits the data perfectly. To reduce the amount of overfitting and increase the smoothness, we relax the constraint that we have to perfectly interpolate. Thus, we use the minimum number of basis functions<sup>10</sup> such that:

$$\sum_{i=1}^n (w_i(y_i - \hat{y}_i))^2 \leq s$$

Advantages	Disadvantages
<ul style="list-style-type: none"> <li>— can be assigned degrees of freedom</li> <li>— extendable to "smooth" version</li> <li>— performs also well if points are not equidistant</li> </ul>	<ul style="list-style-type: none"> <li>— smoothing process does not translate well to a interpretation (unlike smoothing splines)</li> <li>— choice of smoothing parameter <math>s</math></li> </ul>

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. But we also have to assure that  $\alpha$  is big enough.

<sup>10</sup>So we do not require one basis function for each neighboring pair of knots. SciPy uses FITPACK and DFITPACK, the documentation suggests that smoothness is achieved by reducing the number of knots used

313 **3.4.6 Natural Smoothing Splines**

314 Let  $\mathcal{F}$  be the Sobolev space (the space of functions of which the second derivative is  
 315 integrable). Then the unique<sup>11</sup> minimizer

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

316 is a natural<sup>12</sup> cubic spline (i.e. a piecewise cubic polynomial function). The objective  
 317 function ensures that we decrease the curvature while keeping the RMSE low.

Advantages	Disadvantages
<ul style="list-style-type: none"> <li>— Can be assigned degrees of freedom (trace of the hat-matrix).</li> <li>— Efficient estimation (closed form solution).</li> <li>— Intuitive penalty (we don't want the function to be too "wobbly" — change slopes).</li> <li>— Also performs well if points are not equidistant.</li> <li>— Fixes the Runge's phenomenon (fluctuation of high degree polynomial interpolation).</li> </ul>	<ul style="list-style-type: none"> <li>— The tuning parameter <math>\lambda</math> must be chosen. This can be done via cross validation and optimizing a score function (e.g. the RMSE).</li> </ul>

318 **3.5 Tuning Parameter Estimation**

319 Many of the interpolation methods introduced in section ?? and ?? include a free parameter.  
 320 To determine this parameter for a specific interpolation method, we will estimate the  
 321 absolute residuals using OOB estimation and then optimize the parameter using a score  
 322 function. We clarify the procedure step by step:

- 323 i.) Construct a set  $\Lambda$  of candidate parameters that generously covers the parameter  
 324 space.
- 325 ii.) Consider  $\mathcal{P}$ , a set of Pixels.
- 326 iii.) For each parameter  $\lambda \in \Lambda$  consider the individual pixels and compute the LOOCV<sup>13</sup>  
 327 for the absolute residuals of the specific NDVI-interpolation method for all Pixels in  
 328  $\mathcal{P}$  and store them in the set  $R_\lambda$ .
- 329 iv.) Determine  $\lambda_{optimal} = \arg \min_{\lambda \in \Lambda} q_{90}(R_\lambda)$ , where we describe the 90% quantile with  
 330  $q_{90}$ .

331 We choose quantile(90) as our optimization function because we want to allow 10% of  
 332 outliers (corrupt points) but also aim for an accurate fit in 90% of the cases.

333 Figure ?? exemplifies the effect of the optimization function (different quantiles). To  
 334 summarize, we may say that the higher the quantile, the stronger the smoothing.

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

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

<sup>13</sup>For a definition of the leave-one-out-cross-validation we refer to section ??

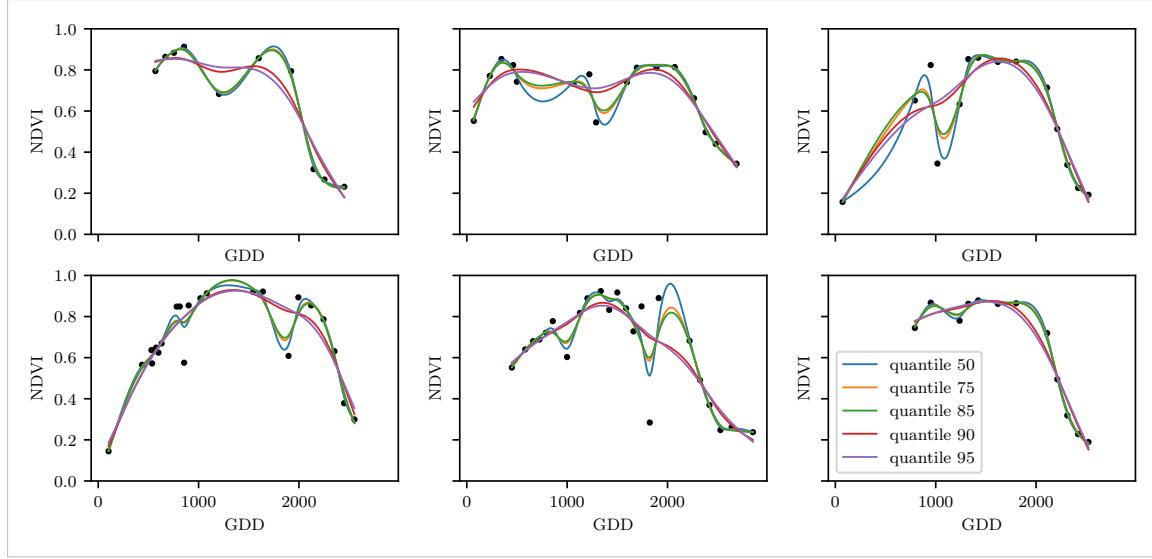


Figure 3.5: Smoothing splines fit with smoothing parameter optimized by minimizing the given quantile of the absolute leave-one-out residuals. Note that the larger the considered quantile is, the smoother the resulting curve becomes.

### 3.6 Robustification

Now we discuss a general approach of how to make an interpolation more robust against outliers. The main idea is to give less weight to observations that have high residuals after the initial (or if we reiterate, the previous) fit.

Even though the procedure is taken from the robust version of the LOESS smoother (cf. section ?? and ?), we can apply it to every interpolation method that allows for prior weighting of observations.

Before we describe the procedure, we define a function that will determine the weight given to each observation, such that observations with large-scaled residuals will have less weight. That is the bisquare function  $B$ :

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

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

$$w_i^{\text{new}} := w_i^{\text{old}} B\left(\frac{|r_i|}{6 \text{ med}(|r_1|, \dots, |r_n|)}\right); \quad r_i := y_i - \hat{y}_i$$

and interpolate again using the new weights. We can iterate this reweighting and stop after several steps or when the change of the values is smaller than some tolerance.

Note that this procedure is indeed robust since we use the median for the normalization which has a breakdown point<sup>14</sup> of 50%.<sup>15</sup>

<sup>14</sup>Intuitively, the breakdown point denotes the fraction of observations a “vicious” player can replace without breaking the estimator. For example, the median has a breakdown point of 50%.

<sup>15</sup>The breakdown point relates only to outliers in the  $y$  values. Note that we do not require the interpo-

348 **3.6.1 Our Adjustment:**

In the case that we would like to apply prior weights, 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

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

349 for  $r, w \in \mathbb{R}^n$ . <sup>16</sup>

350 **3.6.2 Examples and Conclusions**

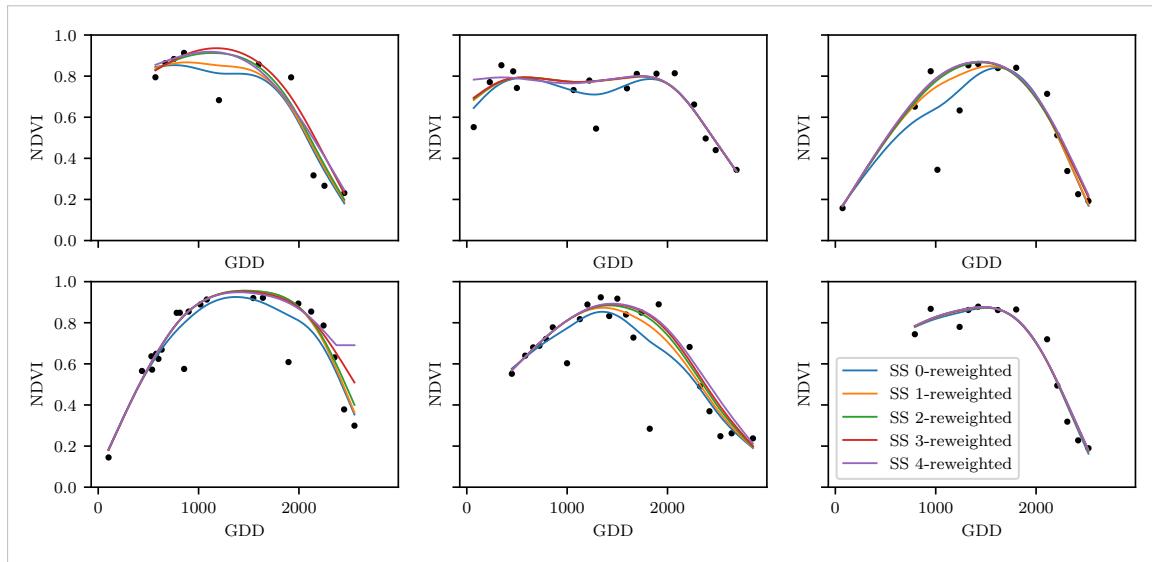


Figure 3.6: Smoothing Splines fitted to different (SCL45) NDVI time series. Iterations of a robustifying refit (as indicated in section ??) are also displayed

351 In figure ?? we observe for six pixels how the NDVI time series interpolated with smoothing  
 352 splines looks after 0, 1, 2, 3, 4 iterations (See appendix figures ??, ??, ?? and ?? for the  
 353 analogous figures of the other interpolation methods).

354 Indeed, we observe how the interpolated time series is less affected by outliers after each  
 355 iteration. We notice the biggest difference in the first iteration. Furthermore, in the plot  
 356 at the bottom left we see how the interpolation ‘escapes’ from the right endpoint with  
 357 each successive iteration, even though our intuition does not necessarily identify this point  
 358 as an outlier. Therefore, in the following, we will always perform only one iteration and  
 359 then stop.

consider  
naming  
the sub-  
plots

360 **3.6.3 Upper Envelope Approach - Penalty for Negative Residuals**

361 If we artificially increase the negative residuals in ?? by multiplying (e.g. factor 2), the  
 362 corresponding points will get less weight in the next iteration. This allows us to create an

lation methods to be robust, since the residual for an outlier will still be larger than for non-outliers and thus will be down weighted more and more in each iteration (because for the next iteration the residual of the outlier will be even larger, since we gave less weight to it).

<sup>16</sup>This adjustment is also necessary to keep the scale estimation meaningful during the iterations.

Table 3.2: Comparing the goodness of fit for different interpolation methods measured with the statistics listed in the left column. 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	FR	$SS^{rob}$	$LOESS^{rob}$	$DL^{rob}$	$BSPL^{rob}$	$FR^{rob}$
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

363 interpolation that resembles an upper envelope. Intuitively, this upper envelope can be  
 364 thought of as a sheet that is laid on top of the points.

365 This approach is based on the premise that we tend to underestimate the NDVI (as in  
 366 REF-savitzky-golay). Since we want to develop a general method that is in principle not  
 367 related to the NDVI, we will not pursue this approach further.

### 368 3.7 Performance Assessment

369 Next, we will benchmark the different interpolation methods with and without robustifi-  
 370 cation. For this, we will use the same technique as we did for the parameter determina-  
 371 tion in section ???. On  $B_\lambda$  we apply the RMSE and different quantiles and present the results  
 372 in table ??.

### 373 3.8 XXX Evaluation

- 375 – ss dominate (i.e. have better benchmark values w.r.t. all considered statistics) b-splines  
 376 (robustified and non-robustified)
- 377 – dl dominate Fourier (robustified and non-robustified)
- 378 – loess slightly dominates ss, but we prefer ss because of the smoothness guarantees (com-  
 379 pare the figures ?? and ??).
- 380 – use dl and ss in the following (keeping robustified and non-robustified variants)

link to  
table  
explaining  
the  
meth-  
ods?

381 **Chapter 4**

382 **NDVI Correction**

383 Let's remind ourselves that the data from the S2 is distributed with an SCL and we  
384 therefore have some information about what is observed at each pixel for each sampled  
385 time (cf. table ??). So far, we have only considered cloud-free points (i.e., SCL-classes 4  
386 and 5). In this chapter, we would like to improve the NDVI interpolation by inspecting  
387 also other SCL-classes and by using more information than just the two bands used to  
388 calculate the NDVI (B4 and B8).

389 **4.1 Considering other SCL Classes**

390 In figure ?? we notice that some blue points which correspond to the SCL-class 10 (thin  
391 cirrus clouds) follow the interpolated line closely and that they might be useful in improving  
392 an interpolation fit.

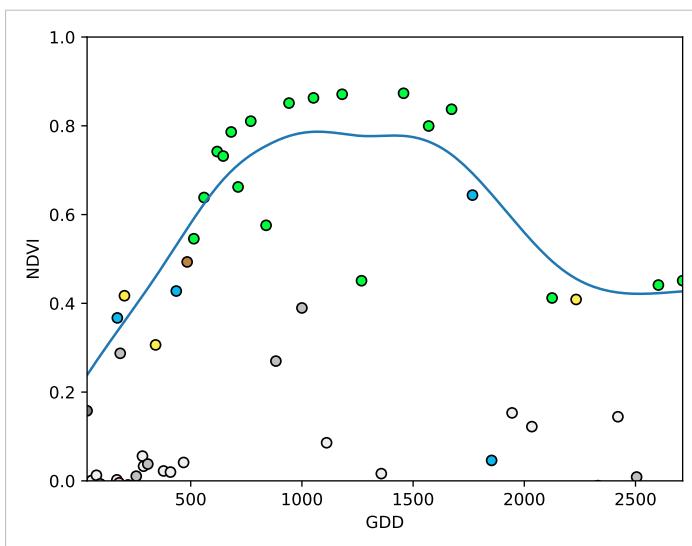


Figure 4.1: A smoothing splines fit considering green and yellow points (SCL45)

393 To get an impression of whether there is some useful information contained in the remaining  
394 SCL-classes (all except 4 and 5) we would like to compare the observed NDVI with the  
395 true NDVI. But since we do not have any ground truth data, we will make the following  
396 assumption:

397 **Assumption 1.** The “true” NDVI value at time  $t$  can be successfully estimated by out-of  
 398 bag (OOB) interpolation using high-quality observations. That is, the interpolated value  
 399 — using an interpolation method from chapter ?? — considering the points  $P^{SCL45} \setminus P_t$ .  
 400 In the following, we will call this estimate the “true”-NDVI.

401 We would like to get an idea if there is any information we can recover from SCL-classes  
 402 other than 4 and 5. For that, we will check for the other SCL-classes if there is a relation  
 403 between the “true” NDVI (derived with Smoothing Splines) and the observed NDVI. Thus,  
 404 we pair each “true” NDVI with its observed one, collect all pairs, and create a scatter plot  
 405 for each SCL-class in fig ???. As expected, the “true” and the observed NDVI seem to be  
 406 highly correlated for SCL45. But we can also detect some patterns of correlation in the  
 407 SCL-classes 2, 3, 7, 8 and 10.

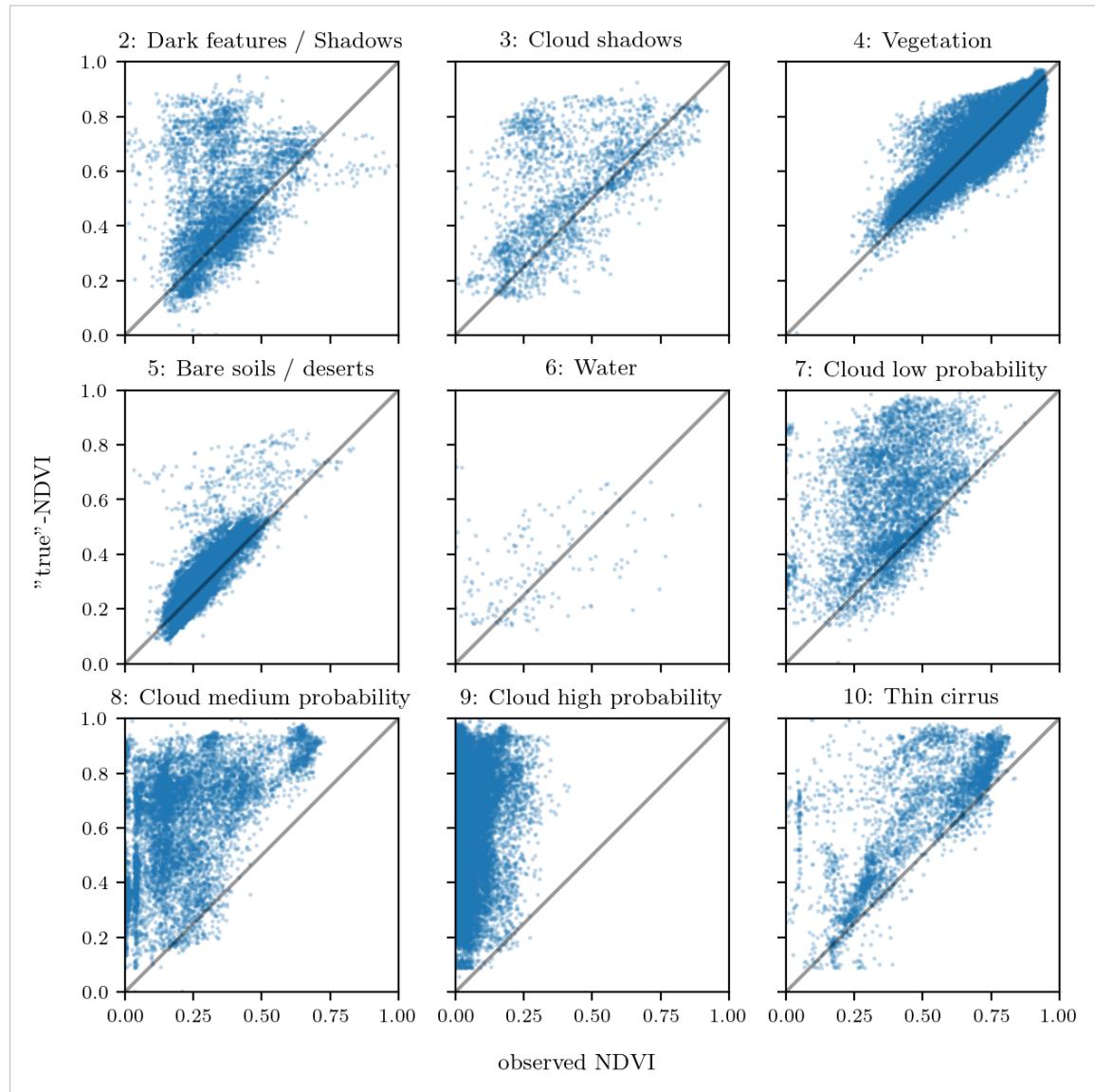


Figure 4.2: For each SCL class, we compare the true NDVI with the observed NDVI. (The true NDVI was estimated with OOB smoothing splines, and we used all observations of 10% of the total training pixels.)

408 It might be tempting to include some of the above SCL classes (for interpolation). But  
 409 on the one hand, the choice would not be objective and on the other hand, the correlation  
 410 seems to be weaker than for SCL45. Therefore, in the following section, we shall try to  
 411 correct the observed NDVI and estimate the uncertainty of each correction.

## 412 4.2 Correction

413 We recall the satellite images in figure ??, where we had cloudy images despite SCL4  
 414 labeled and see fragments in figure ?? even though we are supposed to see clouds (SCL 10  
 415 - Cirrus clouds). The SCL classification is based only on a mixed model trained using the  
 416 s2 bands.

417 We will improve our NDVI interpolation by not relying on the existing SCL classification,  
 418 but by training our own model to estimate/correct NDVI using all S2 bands (see sections ??  
 419 and ??). After we have corrected the observed NDVI, we will assess the uncertainty of  
 420 our corrections and translate it into weights (in section ??). These will be used for the  
 421 subsequent interpolation. This step-by-step procedure is illustrated by the REF graph in  
 422 the appendix.

423 Finally, in section ?? we will evaluate this correction procedure, considering different in-  
 424 terpolation methods and correction models.

hier  
könn-  
te man  
auch  
wieder  
bezug  
nehmen  
auf die  
originale  
Sektion,  
wo man  
SCL  
einführt.  
dort  
fehlt  
mo-  
mentan  
auch  
die Erk-  
lärung,  
dass  
SCL ein  
model  
output  
ist

### 425 4.2.1 Response and Covariates

426 For training an NDVI correction model, we need ground-truth (response) and informative  
 427 covariates. We organize those in a table, where each row corresponds to a  $P_t$  (i.e., a pixel at  
 428 a time  $t$ ). Since ground-truth NDVI data is not available, we will again use the assumption  
 429 ?? and use the “true” NDVI instead. There is no canonical answer to the question of  
 430 which covariates we should use. It is a tradeoff between simplicity/generalizability and  
 431 performance (with the danger of overfitting). Our desire with the NDVI correction is  
 432 to develop a product that is simple for others to understand and use. Therefore, in the  
 433 subsequent, we will only take the spectral data of the satellite (i.e. all the bands) and the  
 434 observed NDVI derived from it as covariates.

### 435 4.2.2 Correction Methods

436 In the following, we will introduce different modelling approaches, which we will use to  
 437 model the relation between the response  $y = y_{\text{true OOB NDVI}} \in \mathbb{R}^n$  and the covariates  
 438 encoded in the design matrix  $X \in \mathbb{R}^{n \times p}$  which contains all covariates.

439 Note that in order to reduce computation time, only 10% of the training data has been  
 440 used to fit the subsequent models which are still more than 120'000 observations.

#### 441 Ordinary Least Squares (OLS)

442 The OLS is a linear model which aims to minimize the sum of the squared residuals. Let  
 443  $y \in \mathbb{R}^n$  be the vector of responses and  $X \in \mathbb{R}^{n \times p}$  be the design matrix, where each row  
 444 corresponds to one pixel and each column consist of one covariate<sup>1</sup>. We assume a linear

<sup>1</sup>Strictly speaking, since SCL-classes are dummy variables

445 relationship between  $y$  and  $X$  and allow for Gaussian noise. That is:

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

446 Assuming that  $X$  is regular, we can estimate the regression coefficients  $\beta$  by

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

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

Advantages	Disadvantages
— Simple method with good interpretability of coefficients.	— Catches only linear relationships. — No integrated variable selection. <sup>2</sup>
— Computationally cheap.	

## 449 LASSO

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

$$\hat{\beta}_\lambda = \arg \min_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 = \arg \min_{\beta \in \mathbb{R}^p \text{ and } \|\beta\|_1 < \lambda} \|y - X\beta\|_2^2. \text{<sup>3</sup>}$$

452 Even though we do not have a closed form solution for equation (??) we can solve it easily  
453 via optimization, since the function  $\beta \in \{\beta \in \mathbb{R}^p | \|\beta\|_1 < \lambda\} \mapsto \|y - X\beta\|_2^2$  is continuous  
454 and convex.

455 ? shows that the LASSO solution tends to be sparse (for moderate  $\lambda$ ). That is  $\beta_i = 0$  for  
456 most  $i = 1, \dots, p$

457 In order to know which  $\lambda$  to choose, we try a huge range of possible values. For each  
458  $\beta_\lambda$ , we calculate the cross-validated  $RMSE_\lambda$ <sup>4</sup> (and its standard deviation  $\sigma_\lambda$  using the  $k$   
459 folds) and define the  $\lambda$  with the smallest corresponding  $RMSE_\lambda$  as  $\lambda_{min}$ . From here we  
460 choose the largest  $\lambda$  for which the  $RMSE_\lambda$  is smaller than  $RMSE_{\lambda_{min}} + \sigma_\lambda$ . This yields  
461 a simpler model while keeping the  $RMSE$  reasonable model.

462 We will apply the Lasso using the selected covariates in section ?? and their second degree  
463 of interactions.<sup>5</sup>

Advantages	Disadvantages
— Usually yields a sparse solution. This tends to give better generalizability (prediction performance on unseen data).	— Estimate is biased. — Computationally expensive.
— Successfully deals with correlation in covariates.	
— Interpretable results.	

<sup>3</sup>The last two terms are equivalent by lagrangian optimization

<sup>4</sup>The cross validated 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  $RMSE$ :

$\sqrt{\sum_{i=1}^n (y - \hat{y})^2 / n}$

<sup>5</sup>This is if our covariates are  $\{a, b\}$ , then we will now use  $\{a, b, ab, a^2, b^2\}$ .

464 **Random Forest (*RF*)**

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

472 To build such a Tree, we will recursively partition the covariate space using greedy splits<sup>6</sup>  
 473 decreasing the RMSE<sup>7</sup> each time. If the set we want to split contains less than a certain  
 474 amount of training points, we stop.

475 To build a *Random Forest* we will bootstrap-aggregate<sup>8</sup> many such Trees<sup>9</sup>. The prediction  
 476 of the Random Forest for a new point  $x$  is then the mean of the predictions from all the  
 477 Trees.

Advantages	Disadvantages
— Captures non-linear relationships.	— The resulting (prediction) function is not continuous but locally constant.
— Captures all interactions and performs automatic variable selection.	— Computationally expensive.
— Can deal with missing data.	— No interpretability.

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

479 A MARS model as introduced in ? can be described by

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

480 where the  $h_m$  are simple functions (explained later) and the  $\beta_m$  are estimated via Least  
 481 Squares.

482 In the building procedure of a MARS model, we first select many of those simple functions  
 483 and later drop some of them to avoid overfitting. For the construction of those simple  
 484 functions, define  $\mathcal{B}$  be the set of pairs of ‘hockystick functions’

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

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

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

---

<sup>6</sup>For computational reasons, we will only use splits along one covariate. So we ‘cut’ our covariate space into rectangles.

<sup>7</sup>To 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).

<sup>8</sup>That is we will sample (with replacement) several times  $n$  observations from our original data and fit a Tree to each such sample.

<sup>9</sup>Building 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. Thus, also the “second best split” can be selected.

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

490 Finally, to avoid overfitting, we prune the set  $\mathcal{M}$  by optimizing a generalized cross valida-  
 491 tion score (GCV).<sup>10</sup>

492 To reduce computational complexity, we follow the recommendation from REF? and re-  
 493 strict  $h$  in equation (??) to be of degree one (so it is also in a pair of  $\mathcal{B}$ ). Consequently,  $\mathcal{C}$   
 494 contains functions with a degree of at most 2.

Advantages	Disadvantages
— Catches non-linear relationships.	— Computationally expensive (can be re- duced by restricting the degree of inter- actions).
— Interpretability via functions in $\mathcal{M}$ and their coefficients.	
— Allows for interactions with variable se- lection.	

#### 495 General Additive Model (*GAM*)

496 GAMs as described in ? are a special case of Projection Pursuit Regression, where only  
 497 the  $p$  directions parallel to the coordinate axes are considered. The result is different to a  
 498 linear model since the coordinate functions are not restricted to be linear but are assumed  
 499 to be non-parametric functions. The model can be written as:

$$g_{add}(x) = \mu + \sum_{i=1}^p g_j(x_j).^{11}$$

500 To estimate the non-parametric functions, we can use smoothing splines (ref sec. ??). For  
 501 this let  $\mathcal{S}_j$  be the function which takes some  $z \in \mathbb{R}^n$  and returns the smoothing splines  
 502 fitted to  $(X_{:,j}, z)$  where the smoothing parameter is optimized by GCV. Since we cannot  
 503 fit all  $g_j$  simultaneously, we will use a strategy named Backfitting. We basically cycle  
 504 through 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  $j = 2, \dots, p$
- $\vdots$

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

Advantages	Disadvantages
— Captures non-linearity.	— No automatic variable selection.
— Good interpretability.	— Computationally expensive.

<sup>10</sup>This 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.

<sup>11</sup>where  $g_j$  is a real-valued function. For identifiability we also demand  $\mathbb{E}[g_j(X_{:,j})] = 0$  for  $j = 1, \dots, p$ .

Ich  
finde die  
sections,  
in denen  
Du die  
Modelle  
erklärest,  
gut.  
Aller-  
dings  
fehlt  
mir die  
Über-  
leitung/Ein-  
leitung,  
warum  
die  
Modelle  
ge-  
braucht  
werden

506

507 

### 4.2.3 Uncertainty Estimation

508 Once we correct the NDVI using the models described in the previous section, we are left  
 509 with the problem that not every correction is equally reliable.<sup>12</sup>. Hence, we are interested  
 510 in a measure of how uncertain an estimate is.

511 We do this by replacing the response with the absolute residuals  $v := |y - \hat{y}|$  and modeling  
 512 their relationship with the covariates defined by  $X$ . In this way, we obtain a model for  
 513 the absolute residuals  $v$  and the estimator  $\hat{v}$ .

514 

### 4.2.4 Interpolation

515 Consider now a pixel  $P$ ,  $\hat{y}^{(P)}$  its corrected NDVI and  $\hat{v}^{(P)}$  the estimated uncertainties of  
 516  $\hat{y}^{(P)}$ . In order to interpolate  $\hat{y}^{(P)}$ , we will give less weight to unreliable observations. Thus,  
 517 we define the weight function:

$$w_\tau^{(P)} := \frac{1}{R} \frac{1}{\hat{v}_\tau^{(P)}}, \quad \text{for } \tau = 1, \dots, n_P$$

518 where  $\tau$  is an index over the satellite images and  $R := \frac{\sum_i^{n_P} \hat{v}_i^{(P)}}{n_P}$  a normalization constant.  
 519 The normalization is needed since for some interpolation methods, inflating the sum of  
 520 weights would decrease the effect of the smoothing.

521 

## 4.3 Resulting Interpolation Strategies

522 We have developed the following procedure to obtain a new interpolation (keyword-wise):

- 523 i.) OOB Interpolation (+ robustify?)
- 524 ii.) Correction
- 525 iii.) Uncertainty estimation
- 526 iv.) Interpolation (+ robustify?)

527 At each step we have a choice, more precisely:

- 528 — Interpolation: Smoothing Splines / Double Logistic
- 529 — Robustify: Yes / No
- 530 — Correction & uncertainty estimation: RF / OLS – considering only SCL-classes /
- 531 — OLS – considering all selected covariates / MARS / GAM / LASSO / no correction.

532 As it is not feasible to try every possible combination, we make the following restrictions  
 533 on which combinations we will consider:

- 534 — We use the same interpolation method each time.
- 535 — Either we robustify both times, or we do not robustify at all.
- 536 — We use the same underlying method for correction and uncertainty estimation.

537 In this fashion, we obtain 28 distinct interpolation strategies, which we will benchmark in  
 538 the next section.

<sup>12</sup>One correction is illustrated in the figure ???. In this figure, the outer points (labeled as clouds) have a large scatter.

539 **4.4 Evaluation Method**

540 In this section, we introduce the relative yield-estimation-accuracy (*RYEA*) and utilize it  
 541 to evaluate the interpolation strategies from section ??.

542 **Definition 4.4.0.1.** (*RYEA*) Let  $y \in \mathbb{R}^n$  be the yield,  $M$  be a model for estimating  $y$ , and  
 543  $\hat{y} = M(X)$  where  $X$  describes the data<sup>13</sup>. We define the *RYEA* as the relative RMSE in  
 544 yield estimation. Formally expressed:

$$\text{RYEA} = \frac{\sqrt{\sum_{i=1}^n (y_i - \hat{y}_i)^2}}{\bar{y}},$$

545 where  $\bar{y}$  denotes the sample mean.

546 **4.4.1 Idea**

547 The fundamental assumption is that the closer the interpolated NDVI time series is to  
 548 the true one, the better it can be used to determine crop yield. Implicitly, we believe that  
 549 an NDVI time series which better models yield will incorporate more true information  
 550 about the underlying vegetation. Therefore, we want to determine a comparable RYEA  
 551 for each interpolation strategy and choose it as a benchmark criterion. This is an objective  
 552 measure, since we have not considered crop yield in any of our previous steps. Moreover,  
 553 this criterion is justified by the fact that yield estimation has been a motivation for the  
 554 interpolation.

555 **4.4.2 Yield Estimation**

556 For all the pixels, we will interpolate the NDVI time series with every interpolation strat-  
 557 egy. From the interpolated NDVI time series, we would like to estimate the yield. However,  
 558 given the high dimensionality and different lengths of the interpolation (not every time  
 559 series has the same start and end point), we must first map each NDVI time series into a  
 560 low-dimensional space. For this, we use the Integral<sup>14</sup> up to the peak Integral<sup>??</sup>  
 561 GDD (i.e. value at which the peak is attained) Integral<sup>??</sup> up to the peak Integral<sup>??</sup>  
 562 after peak Integral<sup>??</sup> from 0-685 GDD Integral<sup>??</sup> from 685-1075 GDD

tabelle  
wäre  
sauberer

564 For the choice we were inspired by REF-kamir. However, we deliberately omit any statistic  
 565 that involves the minimum (e.g. the NDVI-range), since we regard the minimum as a very  
 566 error-prone measure due to the large influence of clouds in the time series.

check  
reference

567 As a result, for each interpolation strategy, a matrix is obtained in which each row corre-  
 568 sponds to a pixel and both the yield and the characterizing statistics are contained. Using  
 569 this matrix, we train a random forest for yield estimation, and compute the integrated  
 570 OOB estimates<sup>15</sup>  $\hat{y}$ . Note that the choice of the modeling approach does not matter much,  
 571 as long as it is general enough (i.e. able to approximate any function) and we use the same  
 572 one for each interpolation strategy. Finally, for each interpolation strategy, we calculate  
 573 the RYEA. The results are shown in table ??.

welche  
charac-  
teriz-  
ing  
statis-  
tics  
genau?

<sup>13</sup>We will use the matrixes derived in section ??

<sup>14</sup>We will only consider the integral of the function  $\max(0, NDVI - 0.3)$ , where 0.3 is assumed to be a minimal NDVI value. REF

<sup>15</sup>By the integrated OOB estimates, we denote the predictions for each pixel where only trees are used, where the pixel has not been used (as  $n_{tree}$ , the number of Trees, grows the fraction of trees which do not contain a certain pixel converges to  $\frac{1}{e}$ ).

574 **Chapter 5**

575 **Results**

576 **5.1 XXX small recap from “Interpolation Methods”**

577 shoud w write 1:1 the sam es in the end of section ??

578 **5.2 Robustification and NDVI-Correction**

579 `var(yield)` 4.034559

580 Min. 1st Qu. Median Mean 3rd Qu. Max. 0.1066 6.1855 7.5595 7.3592 8.7564 13.3508

$$\widehat{\text{NDVI}}_{\text{corr}} = 0.711 \text{NDVI}_{\text{observed}} + 0.215 \mathbb{1}_{SCL=2} + 0.237 \mathbb{1}_{SCL=3} + 0.210 \mathbb{1}_{SCL=4} \\ + 0.116 \mathbb{1}_{SCL=5} + 0.162 \mathbb{1}_{SCL=6} + 0.327 \mathbb{1}_{SCL=7} + 0.474 \mathbb{1}_{SCL=8} \\ + 0.575 \mathbb{1}_{SCL=9} + 0.306 \mathbb{1}_{SCL=10} + 0.512 \mathbb{1}_{SCL=11}$$

$$\widehat{\text{abs}}(\text{NDVI}_{\text{“true”}} - \text{NDVI}_{\text{corr}}) = -0.133 \text{NDVI}_{\text{observed}} + 0.186 \mathbb{1}_{SCL=2} + 0.185 \mathbb{1}_{SCL=3} \\ + 0.146 \mathbb{1}_{SCL=4} + 0.089 \mathbb{1}_{SCL=5} + 0.167 \mathbb{1}_{SCL=6} \\ + 0.203 \mathbb{1}_{SCL=7} + 0.181 \mathbb{1}_{SCL=8} + 0.173 \mathbb{1}_{SCL=9} \\ + 0.180 \mathbb{1}_{SCL=10} + 0.172 \mathbb{1}_{SCL=11}$$

Table 5.1: XXX RMSE of yield prediction

	RF	OLS-SCL	OLS-all	MARS	GAM	lasso	no-correction
ss	1.144	1.033	1.051	1.042	1.046	1.042	1.095
dl	1.150	1.115	1.116	1.116	1.097	1.098	1.159
ss-rob	1.144	1.054	1.084	1.094	1.072	1.071	1.091
dl-rob	1.159	1.128	1.117	1.064	1.093	1.105	1.156

Table 5.2: XXX RMSE of yield prediction

	RF	OLS-SCL	OLS-all	MARS	GAM	lasso	no-correction
ss	0.155	0.140	0.143	0.142	0.142	0.142	0.149
dl	0.156	0.151	0.152	0.152	0.149	0.149	0.158
ss-rob	0.155	0.143	0.147	0.149	0.146	0.145	0.148
dl-rob	0.157	0.153	0.152	0.145	0.148	0.150	0.157

Table 5.3: XXX RMSE of yield prediction

	RF	OLS-SCL	OLS-all	MARS	GAM	lasso	no-correction
ss	nan	nan	nan	nan	nan	nan	nan
dl	nan	nan	nan	nan	nan	nan	nan
ss-rob	nan	nan	nan	nan	nan	nan	nan
dl-rob	nan	nan	nan	nan	nan	nan	nan

581 **Chapter 6**

582 **Discussion**

583 Here in the discussion, you should take up the points you mentioned in the introduction

584 **6.1 Interpolation Methods**

585 You already capture the "main" structure of your thesis with the interpolation and the NDVi correction sections. Can you combine them both in a "synthesis" subsection at the end of the discussion?

586 XXX discuss results from table

587 **6.2 NDVI Correction**

588 **6.2.1 Bootstrap**

589 The question arises if we can build the correction model on the same year as we want to  
590 apply it on. Usually, a similar approach might carry the danger of overfitting. However, we  
591 have not used any ground truth at any point (until the evaluation). Instead, we estimated  
592 the "true" NDVI with the assumption ?? via OOB. Thus, we have bootstrapped our way  
593 out of the problem. Consequently, we reason that we can apply our method to a new  
594 (comparable) dataset and solve the correction again via this bootstrap.

595 **6.2.2 Using Additional Covariates**

597 In section ?? we have only used the spectral data (and the observational NDVI calculated  
598 from them) as covariates. Since we have the weather data available (cf. REF-SEC), it  
599 would be a small effort to incorporate it, together with statistics collected from it (i.e.  
600 GDD or 'rainfall in the last 30 days').

601 We decided against using this data, because on the one hand we have the problem that  
602 we have practically too few observations (we observe only 5 years) and we expect the  
603 weather in our study region to be rather homogeneous which is suggested by the fact  
604 that the weather data published by Meteoswiss are for a grid with a resolution of 1 km.  
605 On the other hand, we want the underlying model not to learn improper relationships.  
606 For example, the model might automatically predict a high NDVI for a day in summer

where  
does  
this sec-  
tion be-  
long to?  
Chapter  
'NDVI  
Correc-  
tion' or  
'Further  
Work'?

607 (detected by high GDD / many sunshine hours / high temperature) just because it is  
608 “used” to observing a lot of vegetation in summer. Including temporally (e.g.,  $P_{t-1}$  and  
609  $P_{t+1}$ ) and geographically adjacent pixels would likely improve performance. However, for  
610 simplicity, we omit it here<sup>1</sup>.

611 - weight/uncertainty function (problem of weight function -> some outer points get really  
612 low weights (just because others in the middle have very little residuals and thus very high  
613 weight))

### 614 6.2.3 High RMSE in Yield Prediction

615 How much can we expect to get? We have multiple sources of uncertainty in the data:

- 616 i.) Uncertainty in Yield data collected by the combine harvester
- 617 ii.) Uncertainty in Yield data through rasterization
- 618 iii.) Uncertainty in satellite images through “measurement errors” introduced via clouds  
619 and other atmospheric effects
- 620 iv.) Uncertainty introduced by interpolating (especially when long data-gaps are present)

---

<sup>1</sup>This is done for simplicity of understanding and using the model, since one would need to adapt to some convention of how to supply the data of adjacent pixels without redundancy (i.e. supplying  $P_t$  multiple times).

621 **Chapter 7**

622 **Conclusion**

623  
624 - itpl methods,  
625 parametric dl  
626 non-param  
627 discarded  
628 kernel methods because of strong bias  
629 kriging because assumptions and highdim parameters  
630 savitzky-golay filter since we will investigate the LOESS which can be thought a  
631 loess slightly best performing itpl method but we notice non-smooth behaviour if  
632 loess > ss > bspl  
633 choose ss because of its meaningful definition (minimizing the integral of the second  
634 - robustifying useful?  
635  
636 XXX draw your conclusion to which you came during this thesis

637 **7.1 Future Work**

638 **7.1.1 Time Series Correction-Interpolation as a General Method**

639 Throughout this thesis, we developed a correction and interpolation method for the NDVI.  
640 However, we never used features of the NDVI. Only the parameter estimated via cross-  
641 validation in chapter ?? depends on the scale of the time series. For simplicity, we could  
642 thus determine the parameter using Generalized Cross Validation (as ? suggests). There-  
643 fore, our approach of interpolation and correction of time series can be applied to arbitrary  
644 time series as long as additional information is available. However, further research is re-  
645 quired, to demonstrate the usefulness of this approach in general.

646 **Example: Cloud Correction with Uncertainty Estimation and Interpolation**

647 This generalization can be used in particular for cloud correction. In the same manner as  
648 we corrected the NDVI time series in chapter ??, we can correct each spectral band and  
649 reunite the corrected bands with the uncertainties. If desired, the time series can also be  
650 interpolated before merging as in chapter ???. The resulting question would be how well  
651 this approach performs.

652 **7.1.2 Minor Improvements**

653 During this project, we also noticed some minor issues that we would have liked to invest-  
654igate further if more resources were available. The most relevant of these are:

- 655 — **Data:** Method how data has been extrapolated to the grid could possibly be improved
- 656 — **Data:** For computational reasons, we mostly considered all years and split the data  
657 (on the pixel level) randomly into a train/test set. A leave one year out cross  
658 validation might yield more accurate results.
- 659 — **Data:** We have not included the spectral bands which have a resolution of 60m. But  
660 precisely these seem to be promising for cloud correction, since they are a proxy of  
661 the water (content and form) in the atmosphere.
- 662 — **NDVI Correction:** Explore the effect of different link functions between the esti-  
663 mated absolute residuals and the weights in section ??.
- 664 — **NDVI Correction:** Yield is not the only target variable of interest. Other variables  
665 like protein content could also be used in section ?? for the method evaluation.

which  
data? I  
assume  
the  
com-  
bine  
har-  
vester  
point  
data?

## 666 Appendix A

# 667 Reproducibility

### 668 A.1 Reproduce Results

669 For reproducibility of the whole computations, we refer to our codebase at:

670 <https://github.com/LGraz/MasterThesis-Code>

671 In order to reproduce our computations and results, set up the directory as described  
672 in the README and execute the computations via `./shell_scripts/reproduce.sh`  
673 and do not execute the python and R scripts by hand (unless you follow the order in  
674 `./shell_scripts/reproduce.sh`).

### 675 A.2 R-Package

676 We also provide an R package for a general time series correction and interpolation if  
677 additional data is available at:

678 <https://github.com/LGraz/CorrectTimeSeries>

679 In our case we consider the NDVI time series and the additional data consists of the unused  
680 spectral bands.

681 We recommend installing it via the `devtools` package by:

682 `devtools::install_github("LGraz/CorrectTimeSeries")`

683 In the following, we shall give a stand-alone example of how the R package can be used:

```
684
685 1 library(CorrectTimeSeries)
686 2
687 3 # load a list of dataframes, each one describes one pixel with the covariates and
688 4 # the response
689 5 data(timeseries_list)
690 6 str(timeseries_list[[1]])
691 7
692 8 # Train/Load RF
693 9 train_model_myself <- TRUE
694 10 if (train_model_myself){
695 11   # Add "true" NDVI (or generally the response), by Out-Of-Bag estimation
696 12   timeseries_list <- lapply(timeseries_list, function(df) {
697 13     df$oob_ndvi <- OOB_est(df$gdd, df$ndvi_observed) # gdd is the time-axis
698 14     df
699 15   })
700 16   # Train correction model
701 17   formula <- "oob_ndvi ~ B02+B03+B04+B05+B06+B07+B08+B8A+B11+B12+scl_class"
702 18   RF <- train_RF_with_fromula(formula, timeseries_list, robustify=TRUE)
703 } else {
```

```
704 19  data(RF_for_NDVI)
705 20  RF <- RF_for_NDVI
706 21 }
707 22
708 23 # ADD CORRECTION
709 24 timeseries_list <- lapply(timeseries_list, function(df) {
710 25   df$corrected_ndvi <- randomForest:::predict.randomForest(RF, df)
711 26   df
712 27 })
713 28
714 29 # Get interpolation for each timeseries
715 30 newx <- 1:1000
716 31 lapply(timeseries_list, function(df){
717 32   ss <- smoothing_spline(df$gdd, df$corrected_ndvi)
718 33   predict(ss, newx)$y
719 34 })
```

Example of how to use the `CorrectTimeSeries` package

721 **Appendix B**

722 **Further Material**

723 **B.1 Interpolation**

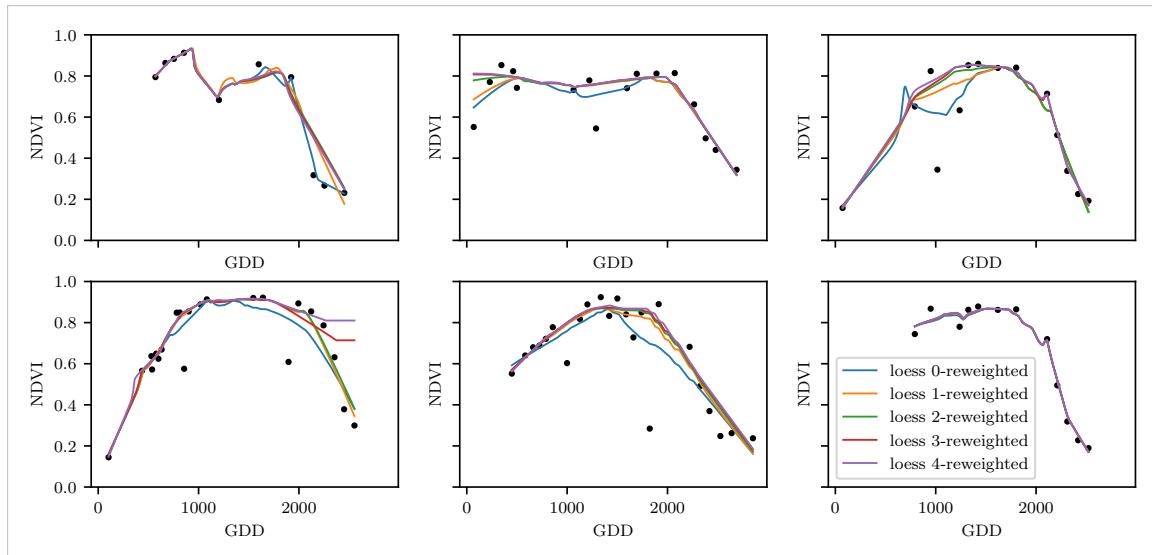


Figure B.1: The LOESS smoother fitted to different (SCL45) NDVI time series. Iterations of a robustifying refit (as indicated in section ??) are also displayed

724 **B.2 NDVI correction**

725 page breaks

```
726
727 1 Call:
728 2 lm(formula = (paste(response, " ~ ", "ndvi_observed + scl_class")),
729 3   data = ndvi_df)
730 4
731 5 Residuals:
732 6   Min     1Q   Median     3Q    Max
733 7 -0.7997 -0.0717  0.0039  0.0695  0.6632
734 8
735 9 Coefficients:
736 10            Estimate Std. Error t value Pr(>|t|)
737 11 (Intercept) 0.21465   0.00230  93.46 < 2e-16 ***
738 12 ndvi_observed 0.71116   0.00346 205.65 < 2e-16 ***
```

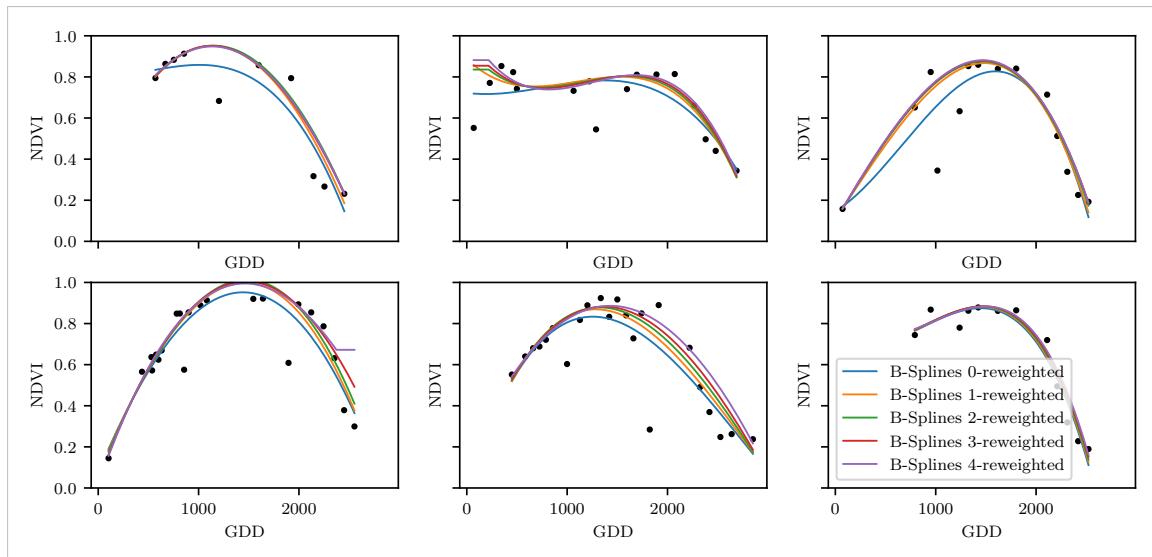


Figure B.2: B-Splines fitted to different (SCL45) NDVI time series. Iterations of a robustifying refit (as indicated in section ??) are also displayed

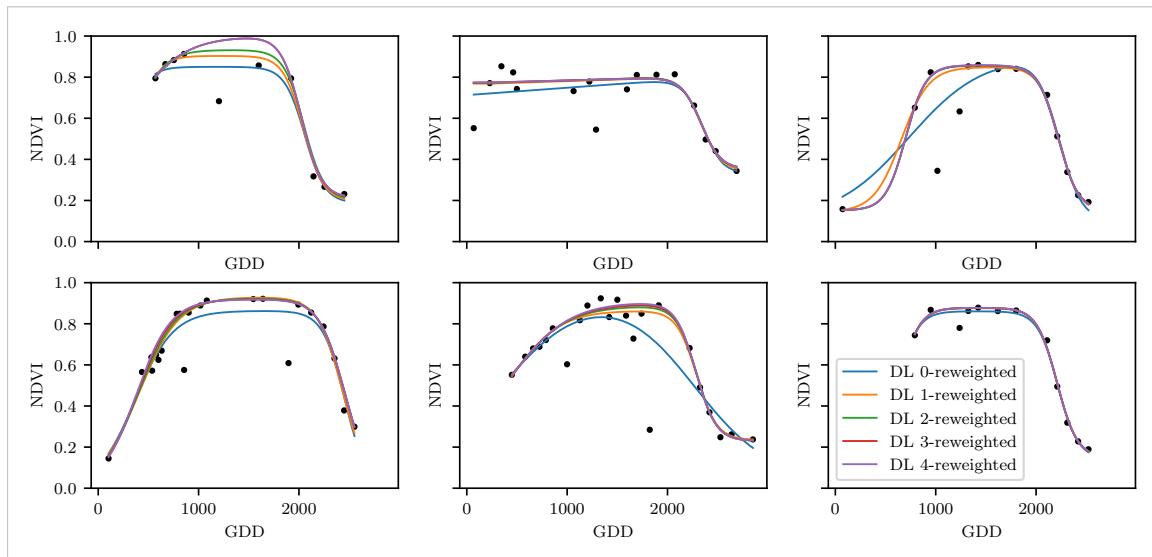


Figure B.3: A Double Logistic curve fitted to different (SCL45) NDVI time series. Iterations of a robustifying refit (as indicated in section ??) are also displayed

```

739 13 scl_class3      0.02205    0.00356    6.20   5.8e-10 ***  

740 14 scl_class4     -0.00431    0.00251    -1.72   0.085 .  

741 15 scl_class5     -0.09875    0.00234    -42.15  < 2e-16 ***  

742 16 scl_class6     -0.05301    0.01104    -4.80   1.6e-06 ***  

743 17 scl_class7     0.11245    0.00274    41.09  < 2e-16 ***  

744 18 scl_class8     0.25963    0.00253    102.57 < 2e-16 ***  

745 19 scl_class9     0.35994    0.00236    152.47 < 2e-16 ***  

746 20 scl_class10    0.09091    0.00308    29.54  < 2e-16 ***  

747 21 scl_class11    0.29784    0.00392    76.06  < 2e-16 ***  

748 ---  

749 23 Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  

750 24  

751 25 Residual standard error: 0.146 on 124978 degrees of freedom  

752 26 Multiple R-squared:  0.532,          Adjusted R-squared:  0.532

```

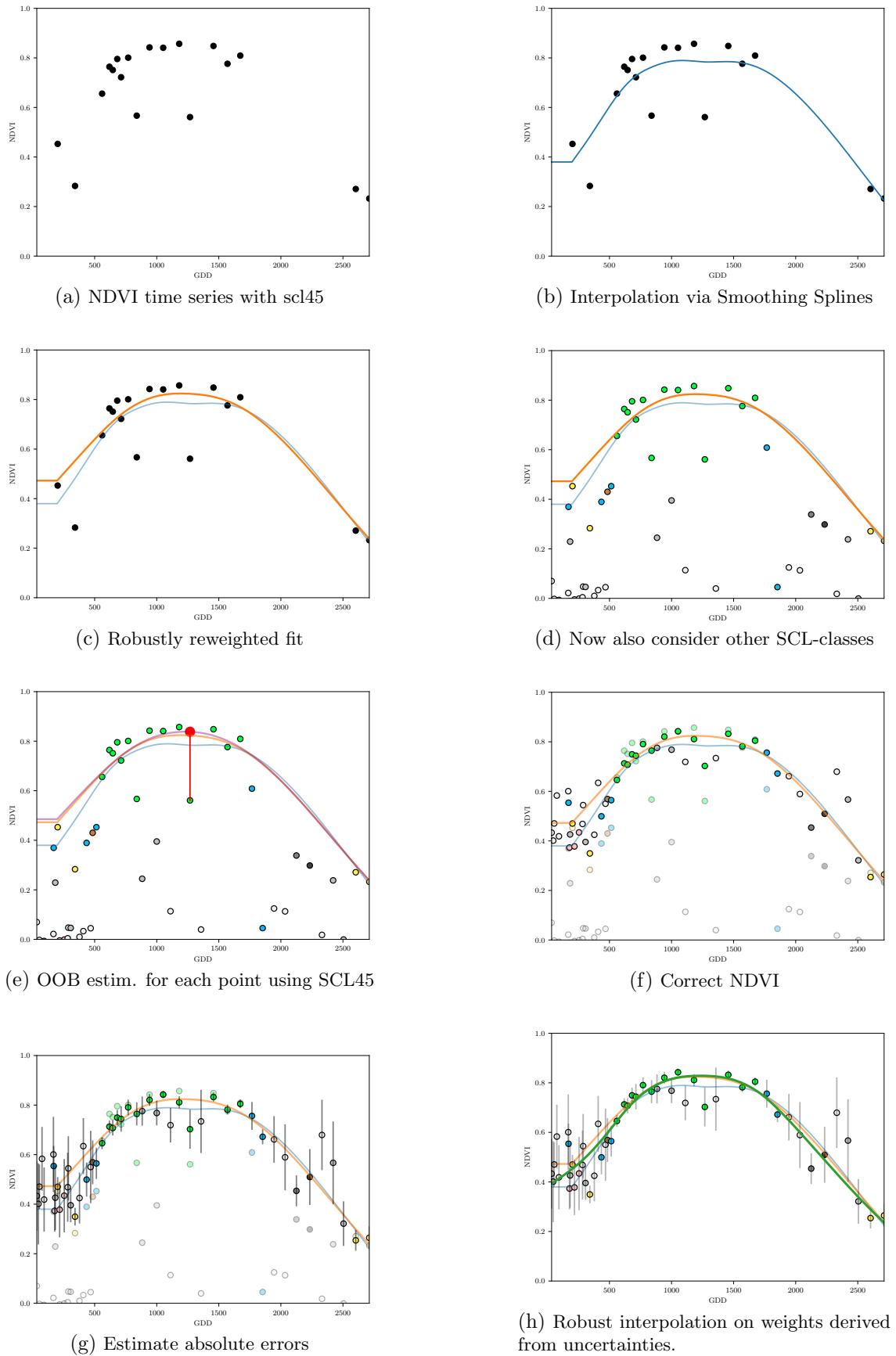


Figure B.4: Stepwise illustration of robust NDVI-Correction. For the color encoding of the SCL classes we refer to table ??.

753 27 F-statistic: 1.42e+04 on 10 and 124978 DF, p-value: <2e-16

R Summary of the NDVI correction model (c.f. equation ??)

```

755
756 1 Call:
757 2 lm(formula = (paste(get_res(), " ~ ", "ndvi_observed + scl_class")),
758 3   data = ndvi_df)
759 4
760 5 Residuals:
761 6   Min     1Q   Median     3Q    Max
762 7 -0.2051 -0.0427 -0.0074  0.0329  0.6589
763 8
764 9 Coefficients:
765 10            Estimate Std. Error t value Pr(>|t|)
766 11 (Intercept) 0.18647  0.00126 147.74 < 2e-16 ***
767 12 ndvi_observed -0.13265  0.00190 -69.80 < 2e-16 ***
768 13 scl_class3 -0.00180  0.00196 -0.92  0.3587
769 14 scl_class4 -0.04069  0.00138 -29.55 < 2e-16 ***
770 15 scl_class5 -0.09698  0.00129 -75.32 < 2e-16 ***
771 16 scl_class6 -0.01906  0.00606 -3.14  0.0017 **
772 17 scl_class7  0.01641  0.00150 10.91 < 2e-16 ***
773 18 scl_class8 -0.00560  0.00139 -4.02 5.7e-05 ***
774 19 scl_class9 -0.01384  0.00130 -10.67 < 2e-16 ***
775 20 scl_class10 -0.00690  0.00169 -4.08 4.5e-05 ***
776 21 scl_class11 -0.01446  0.00215 -6.72 1.8e-11 ***
777 22 ---
778 23 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
779 24
780 25 Residual standard error: 0.08 on 124978 degrees of freedom
781 26 Multiple R-squared: 0.352, Adjusted R-squared: 0.352
782 27 F-statistic: 6.8e+03 on 10 and 124978 DF, p-value: <2e-16

```

R Summary of the NDVI correction model (c.f. equation ??)