0.1 Splines and Interpolation

0.1.1 Interpolation

Interpolation is a tool that can be used - and misused - to extract more information out of a given set of data. Used correctly, interpolation can reveal more information than is initially available and disclose connections not apparent at first, but used incorrectly, it can be manipulated to infer misleading correlations and lead to inaccurate conclusions. Thus it is a tool that must be used with care. Aiming to avoid incorrect deductions and inferences one should at first gain as much knowledge about the data at hand as possible. By understanding how the data have come about and gaining knowledge about the underlying physical theories a somewhat deficient data set can robustly and securely be interpolated to accommodate the needs of the analysis. In the case of this thesis, both knowledge about data gathering and the physics at play have been gained and thus some of the common fallacies may be avoided. The limits of the data available is due to the discrete sampling, leading to a minimum sampling of about 26 samples per meter of ice. When considering that the depth series of 32 years between Tambora and Laki is just above 10 meters, this means that each meter of ice needs to contain at least three years on average. 26 samples per three years might not sound as a bad sampling interval, but if the goal is to show seasonality and give a best estimate of annual layer thickness, interpolation could be put to good use to be able to give better estimates of the exact placement of peaks and valleys.

Existence, Uniqueness and Conditioning

Considering any attempt to create an interpolant to fit a number of data points, the questions of uniqueness and existence is a matter of matching the data points with the number of parameters in the interpolant. If there are too few parameters, the interpolant does not exist, as it will not pass through all data points. If there are too many, the interpolant will not be unique. Formally this can be described through a system of linear equations.
For any data set consisting of $(t_i, y_i),\ i = 1, ..., m$ points, an interpolant can be chosen from a function space spanned by some suitable set of basis functions, $\phi_1(t), ..., \phi_n(t)$. The interpolant can then be described as a linear combination of these basis functions:

$$f(t) = \sum_{j=1}^{n} x_j \phi_j(t) \tag{1}$$

The interpolant can then be found by determining the parameters x_j by re-

quiring that the interpolant f must pass through the M data points (t_i, y_i):

$$f(t_i) = \sum_{j=1}^{n} x_j \phi_j(t_i) = y_i, \quad i = 1, ..., m \tag{2}$$

This can of course also be written compactly in matrix form as a system of linear equations:

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{y} \tag{3}$$

In this equation $\boldsymbol{A}$ is the $m \times n$ basis matrix, which entries consists of the value of the n basis functions evaluated at the m data points, $a_{ij} = \phi_j(t_i)$, the m vector $\boldsymbol{y}$ consists of the known data values y_i, and the n vector $\boldsymbol{x}$ consists of the unknown, to be determined, parameters x_j.
From linear algebra we know, that if we choose the number of basis function ot be equal to the number of data points, $n = m$, the basis matrix will be square, and thus - given the matrix is nonsingular - the system will be determined, and the data points can be fit exactly. Though in some problems it is beneficial to choose the system to be either overdetermined(less parameters than data points, the data cannot be fit exactly) or underdetermined(more parameters than data points, giving freedom to allow satisfaction of additional properties or conditions).
So the existence and uniqueness of an interpolant is given by the non-singularity of the basis matrix, be it square or not and the conditioning of the matrix points to the parameters' sensitivity to perturbations. An ill-conditioned basis matrix will lead to high sensitivity in the parameters, but this problem can still be approximately solvable through Gaussian elimination with partial pivoting, but this solution will mean that the coefficients may be poorly determined.

Polynomial Interpolation

The most common way to determine an interpolant is through polynomials. Denoting a set of all polynomials of degree at most $k, \quad k \geq 0$ as $\mathbb{P}_k$, it can be seen that this set forms a vector space of dimension $k+1$. The basis functions that span this vector space can be chosen to be composed of a number of different functions and this choice has a great influence on both the cost of computation and manipulation of the interpolant, and the sensitivity of the parameters, i.e. the conditioning of the basis matrix.
Considering n data points it is obvious to choose $k = n-1$ so that the dimension of the vector space matches the number of data points. The maybe most

natural choice of basis for $\mathbb{P}_{n-1}$ is one that consists of the first n monomials[1],

$$\phi_j(t) = t^{j-1}, \quad j = 1, ..., n. \tag{4}$$

Thus any given polynoial $p_{n-1} \in \mathbb{P}_{n-1}$ will be of the form

$$p_{n-1}(t) = x_1 + x_2 t + \cdots + x_n t^{n-1}. \tag{5}$$

In this basis the system of $n \times n$ linear equations will be of the form

$$\boldsymbol{A}\boldsymbol{x} = \begin{bmatrix} 1 & t_1 & \cdots & t_1^{n-1} \\ 1 & t_1 & \cdots & t_1^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & t_1 & \cdots & t_1^{n-1} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \boldsymbol{y}. \tag{6}$$

This type of matrix with geometric progression, i.e. the columns are successive powers of some independent variable t is called a Vandermonde matrix. When using the monomial basis and using a standard linear equation solver to determining the interpolants coefficients requires $\mathcal{O}(n^3)$ work and often results in ill-conditioned Vandermonde matrices $\boldsymbol{A}$, especially for high-degree polynomials. This ill-conditioning is due to the monomials of higher and higher degree being more and more indistinguishable from each other. This makes the columns of $\boldsymbol{A}$ nearly linearly dependent, resulting in almost singular matrices, and thus highly sensitive coefficients. For high enough n, the Vandermonde matrix becomes efficiently singular, to computational precision at least, though, as mentioned, this can be worked around, but requires some additional computational work.

REFERENCE!!!

Maybe an illustration here? See pp. 314 in Scientific Computing.

Piecewise Polynomial Interpolation and Splines

The amount of work needed to solve the system as well as the conditioning of the system can be improved by using a different basis all together. Some different bases superior to the monomial that are worth mentioning are the Lagrange basis functions, the Newton basis functions and the orthogonal polynomials. But for this thesis we take a step further into the interpolation theory, as the choice of basis functions might not be enough to work around some of the problems connected with fitting a single polynomial to a large number of data points(i.e. oscillatory behaviour in the interpolant, nonconvergence or issues around the boundaries).

REFERENCES!!

[1] Roughly speaking, a polynomial with only one term.

These practical and theoretical issues can be avoided through the use of piecewise polynomial interpolation, with the advantage that a large number of data points can be fitted with low-degree polynomials.
When turning to piecewise polynomial interpolation of the data points (t_i, y_i), $i = 1, ..., n$, $t_1 < t_2 < \cdots < t_n$, a different polynomial is chosen for each subinterval $[t_i, ti + 1]$. Each point t_i, where the interpolant changes is called knots or control points. The simplest piecewise interpolation is piecewise linear interpolation, where each knot is connected with a straight line. If we consider this simple example it appears that by eliminating the problems of nonconvergence and unwanted oscillatory behaviour, the smoothness of the interpolant is sacrificed. This might be true for this simplistic example but since there are a number of degrees of freedom in choosing each piecewise polynomial interpolant, the smoothness can be reintroduced by explotiting a number of these measures. One way of doing this is by demanding knowledge of both the values and the derivatives of the interpolant at each data pint. This just adds more equations to the system, and thus to have a well-defined solution, the number of equations must match the number of parameters. This type of interpolation is known as Hermite interpolation. The most common choice for this interpolation, to still maintain simplicity and computational efficiency, is cubic Hermite interpolation. This introduces a piecewise cubic polynomial with n knots, and thus $n-1$ interpolants each with 4 parameters to fit, leading to $4(n-1)$ parameters to be determined. Since each of the $n-1$ cubics must match the data points at each end of the subinterval, it results in $2(n-1)$ equations, and requiring the derivative to be continuous, i.e. match at the end points, an additional of $n-2$ equations are taken in. This leads to a system consisting of $2(n-1)+(n-2) = 3n-4$ equations to fit to the $4n-4$ parameters. This leaves n free parameters, meaning that a cubic Hermite interpolant is not unique and the remaining free parameters can be used to accommodate further or additional constraints that might be around the problem at hand.

Cubic Spline Interpolation

A spline is a piecewise polynomial of degree k that is continuously differentiable $k-1$ times.

One way of using the remaining free parameters is by introducing *splines*. A cubic spline is, given the spline definition, a piecewise cubic polynomial, a polynomial of degree $k = 3$, and must then be $k - 1 = 2$ times differentiable. Thinking back on the Hermite cubic, we were left with n free parameters. By demanding continuity of also the second derivative, we introduce $n-2$ new parameters, leaving only 2 final parameters to be free. These 2 remaining parameters can be fixed through a number of different requirements, e.g. by forcing the second derivative at the endpoints to be zero, which leads to the *natural* spline.
The Hermite and spline interpolations are useful for different cases. The Her-

mite cubic might be more appropriate for preserving monotonicity if it is known that the data are monotonic. On the contrary, the cubic spline may enforce a higher degree of smoothness as it takes the second derivative into account as well.

0.2 Peak Detection

Write this entire section

0.3 Parallelization

Do some actual parallelization! And write this entire section