Assessment and reduction of climate model errors using data assimilation

Kaj-Ivar van der Wijst*
Mathematical Institute, Utrecht University, 3508 TA, Utrecht, The Netherlands

Bachelor thesis

Under supervision of Svetlana Dubinkina
CWI, 1090 GB Amsterdam, The Netherlands

June 2016

Abstract

During this thesis, we develop three methods to assess the error of a climate model by comparing it to observations. The first consists of averaging over a chosen spatial region, and comparing the resulting time series. The second method uses Empirical Orthogonal Function decomposition to obtain a time-dependent (Principal Component) and space-dependent (EOF) component of the model output and observations. We then fix the spatial component by projecting one dataset onto the EOF of the other. This yields two Principal Components which we can compare. The third method uses Kalman filter data assimilation to obtain an estimate for the variance of the model output. We also implement a rudimentary form of performing data assimilation directly on an EOF.

1 Motivation of the thesis

We live in a world where technological advancements, globalization, geopolitical decisions and many other factors continuously influence our ways of life. Because of this, temperatures are changing, water levels are rising and extreme weather events occur more frequently than in the pre-industrial era, as described in the IPCC report (Cubasch et al., 2013). In order to be better prepared for these climate changes, it is important to know *how* the climate will change. These climate predictions form the basis for policy makers all around the world.

To be able to predict the future climate, we also need a good knowledge of the past climate. This gives an insight in the long term climate variations and the possible natural and human sources of these changes. Global temperature changes from a long time ago can have a big impact on future climate, through various periodic and feedback processes. Another reason to consider past climate is that it can be assessed using measurements, whereas the future climate is always a prediction.

The need for accurate representations of past climate has triggered the development of climate models that range in time scales from days, to years all the way to millions of years in the past. All of these have different uses and applications. These models use the power of modern computers and knowledge from physics, chemistry, life sciences, geosciences and more to produce representations of various quantities, like temperature and humidity of the past. However, these models are not perfect. The spatial area and time component they use have to be discretized, which could discard small scale effects. Typically, these models use approximations of the true physical systems, since for example, the full Navier-Stokes equations, describing the way continuous fluids behave, cannot be solved exactly. Another source of error is the initial condi-

*Corresponding author. E-mail: k.vanderwijst@uu.nl

tions: these often have to be estimated using either available measurements or a certain physical balance state. All these approximations add to the error of the model. In order to properly interpret the results of such models, it is important to know these errors. They are, however, generally very difficult to deduce from the model itself. Other methods of assessing the errors need to be used.

Once the errors of a model are estimated, we try to reduce the errors with respect to the true state (true temperature, true humidity, etc.) by combining the model with some available observations. This process is called *data assimilation*.

In this thesis, we will discuss various methods to assess the errors of a climate model, the LOVECLIM model, by comparing the output of the model to a set of temperature measurements of the last 150 years, the HadCRUT dataset. Moreover, we will reduce these errors by combining the model to various sets of observations.

2 Used model and measurements

2.1 LOVECLIM model

The model we use for this thesis, is the LOVECLIM model used to reconstruct climate of the past. LOVECLIM incorporates changes in the atmosphere, ocean and sea ice, land surface and vegetation, ice sheets, icebergs and the carbon cycle, as described by Goosse et al. (2010). The model evolves physical quantities like atmosphere temperature, ocean temperature, atmospheric CO_2 concentration, vegetation coverage and more. Temperature, however, is sensitive to most of these quantities (e.g., CO_2 concentration: Knutti & Hegerl (2008), vegetation coverage: Lloyd & Taylor (1994)). This thesis focuses therefore solely on the temperature reconstruction of the model. Since LOVECLIM is a coupled model, meaning that all variables influence one another, we don't have to consider the temperature of the whole atmosphere and ocean: it is enough to use the temperature of the bottom layer of the atmosphere only.

As shown in table 1, the model computes temperatures on a 64×32 spatial grid of the whole world (resolution of 5.625°). In climate research, two sets of temperatures are usually compared by either looking at their mean, or by looking at their *anomalies*: these are relative temperatures compared to a specific time average. The model uses regular temperatures, whereas the measurements are given in anomalies. This is discussed in more detail in section 3.2.

Grid size:	5.625°, 64×32 grid
Times:	monthly, 01-1849 to 12-1998
Unit:	temperatures, in °C

Table 1: LOVECLIM model*: the output is temperature on a* 64×32 *grid.*

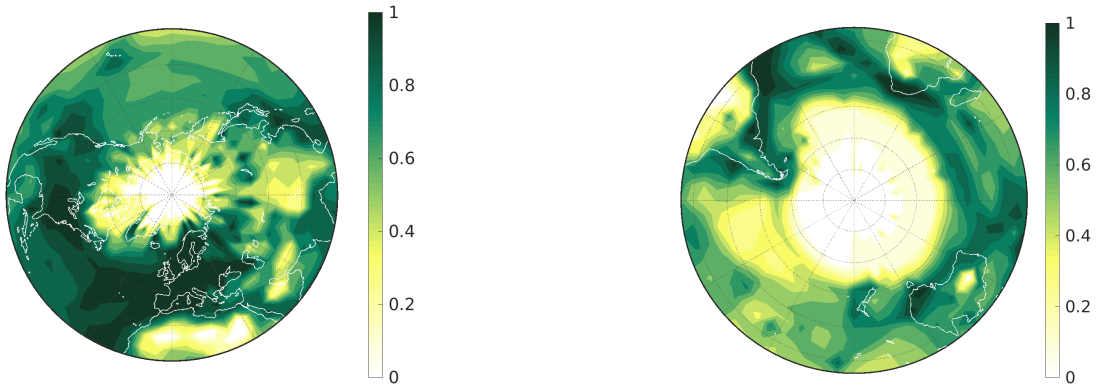
2.2 HadCRUT Measurements

In order to test the quality of the temperature reconstruction of the model, it has to be compared to measurements. These are available in the HadCRUT dataset, as described by Brohan et al. (2006). This is a combination of temperature measurements from land stations and marine data, presented as anomalies calculated with respect to 1961 to 1990.

The HadCRUT4 dataset features the anomalies on a 72×36 world wide grid, with measurements starting from January 1850, as seen in table 2. However, the dataset is incomplete due to missing measurements, with 85% missing values in 1850 to about 30% missing values in 2000. The spatial coverage, defined as the percentage of time moments with existing values (not `nan`) per grid point, is plotted in Fig. 1a and 1b for the North and South pole respectively. We project these using the Breusing Harmonic Mean Projection (Lambers, 2016). The time coverage (percentage of grid points with existing values per time moment), is plotted in Fig. 2.

Grid size:	5°, 72×36 grid
Times:	monthly, 01-1850 to 09-2015
Unit:	anomalies, with respect to 1961-1990, in °C

Table 2: HadCRUT4 measurements*: the measurements are fitted on a* 72×36 *grid, with anomalies calculated from the monthly average of 1961 to 1990.*



(a) *Northern hemisphere* **(b)** *Southern hemisphere*

Figure 1: Coverage of the HadCRUT dataset. *The coverage is calculated for every grid point as the number of existing monthly measurements divided by the total number of months between 1850 and 1998. The Northern Hemisphere has much more measurements in Europe and North America, while the Southern Hemisphere has more in South America, Australia and parts of the Pacific Ocean. A coverage of 0 means no measurements at all during all these years, up to 1 where every month has an existing measurement.*

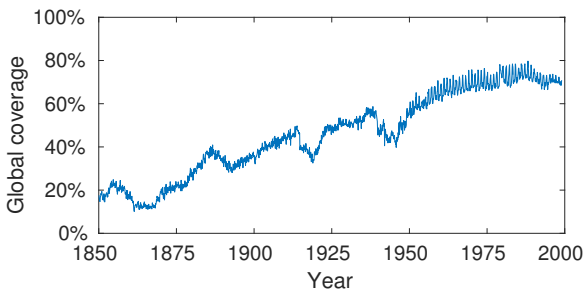


Figure 2: *Global coverage of HadCRUT dataset for every month from 1850 to 1998.*

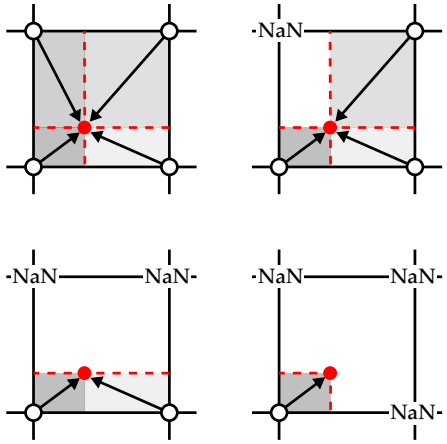


Figure 3: *Hybrid interpolation. The value at the new grid point is a weighted average of either all four, three, two or one existing neighbouring values.*

3 Preparation of data

As it can be seen in tables 1 and 2, some steps have to be taken to properly compare the temperature of the model with the measurements. First, the measurements on the 72×36 grid should be fitted to match the model grid (64×32) using interpolation. Finally the temperatures in the model should be converted to anomalies with respect to 1961 to 1990.

3.1 Interpolation

The temperatures from the LOVECLIM model were computed on a 64×32 grid, corresponding to a grid distance of $5.625°$. The HadCRUT4 measurements, however, used a 72×36 grid, which has a grid distance of $5°$.

First we have to extend the computational domain by ghost grid points to account for the fact that temperatures at longitude $0°$ correspond to temperatures at longitude $360°$. Since we only use (bi)linear interpolation, one ghost grid point at every end of the longitude is enough: one at $-5.625°$, one at $366.625°$, for every latitude.

3.1.1 Nearest neighbour interpolation

The easiest interpolation method is nearest neighbour interpolation. Every value on the new grid gets the value of the nearest original grid point. This method is very rough: it is essentially a zeroth-order Taylor approximation in 2D.

3.1.2 Hybrid bilinear interpolation

A widely used interpolation method in climate science is bilinear interpolation, as described by Accadia et al. (2003). Every interpolated value is equal to a weighted average of the four nearest original grid points. This method, however, only works if all neighbouring grid points exist (a missing value is shown by a NaN-value, not-a-number).

We have therefore chosen for a hybrid bilinear interpolation method where the value of every new grid point is a weighted average of all the existing neighbouring old grid points. The closer an existing old point is to the new point, the bigger its weight in the average. There are four cases, visualized in Fig. 3.

3.1.3 Comparison of interpolation methods

When calculating the average difference between the two kinds of interpolation, we come very close to zero (the average difference over 148 years is $0.0035°C$). However, locally, differences can grow up to $2°C$ to $3°C$. This is shown for one month (January 1975) in Fig. 4. These are significant differences, which show how important it is to use a more subtle, refined interpolation method, like the hybrid (bi)linear interpolation explained above.

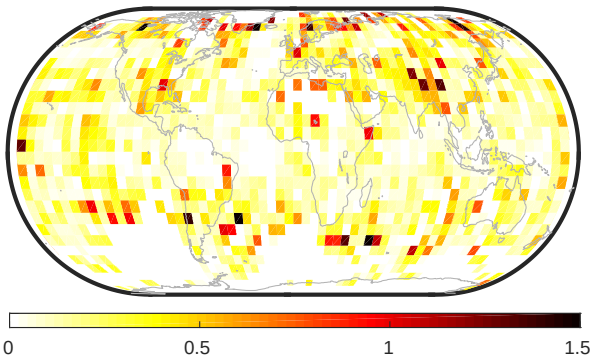


Figure 4: *Difference in °C between (bi)linear interpolation and nearest neighbour interpolation, in January 1975. The average is very close to 0, with maximum differences of about* $2°C$.

3.2 Calculating anomalies of the temperature of the model

As mentioned previously, two sets of temperatures can be compared by either looking at their averages, or at their anomalies. Anomalies are relative temperatures, compared to the average over a given period. In this thesis, we have chosen to only work with anomalies. Since the model outputs absolute temperatures, we have to transform them. For this, we use the same time period as the anomalies of the HadCRUT measurement, namely 1961 to 1990.

The anomalies are obtained by first calculating the average per month over this time period. This gives a set of 12 average temperatures, which are removed from all the corresponding months in the temperature of the model, giving the anomalies with respect to 1961 to 1990. In Fig. 5, we show the reference temperatures for the temperature of the model (solid lines), and the HadCRUT measurements (dashed lines), averaged over Europe, the Northern Hemisphere and the Southern Hemisphere. The monthly averages of the Northern and Southern hemispheres for the model and data anomalies are close to each other, but there is an almost constant $5°C$ difference for Europe. This shows that it would also be interesting to compare the means of the two temperature sets, instead of only the anomalies.

4 Space averaged comparison

A first indicator on the quality of the temperature reconstruction of the model is obtained by comparing the average of various regions with the average of the measurements over the same regions. Our analysis consists of looking at 5 year and 20 year averages for Europe, the whole Northern hemisphere and the whole Southern hemisphere separately. A measure of skill is given by the respective correlation coefficients.

We also use a second measure of skill, the RMS

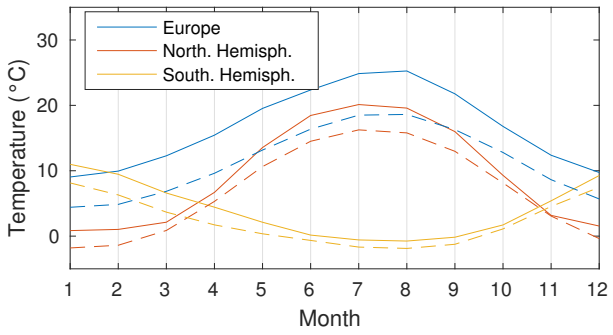


Figure 5: *Reference temperatures used to calculate the anomalies. These have been averaged over Europe, the whole Northern hemisphere and the Southern hemisphere. Month 1 is January. The solid lines correspond to temperatures of the model, dashed lines of the HadCRUT dataset.*

error. This is defined by:

$$RMSE := RMS(\mathbf{w}^{\text{model}} - \mathbf{w}^{\text{obs}}) = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(w_t^{\text{model}} - w_t^{\text{obs}})^2}, \quad (1)$$

where $\mathbf{w}^{\text{model}}$ and $\mathbf{w}^{\text{obs}}$ are the time series of the model output and observations respectively, obtained after averaging over a given spatial area. n is the number of time steps.

4.1 Europe

We have chosen to consider the temperatures over Europe on a rectangular box from $60°N$ to $35°N$ and $15°W$ to $30°E$.

As shown in Fig. 6a and 6b, the temperature of the model shows rather good accordance with the measurements for the last 50 years. This is even better visible in the 20 year average. The trend is still accurate for the years between 1850 and 1930, yielding a correlation coefficient of $\rho = 0.86$ for the 20 year average, and an RMS error of $0.11°C$.

However, on the 5 year time scale, we see many differences. For example, the period of 1940-1945 was much colder than shown by the model. This is because the model apparently doesn't account for the extreme harsh winter of 1940 (Rijkswaterstaat, 1942).

4.2 Northern Hemisphere

Averaging over the whole Northern Hemisphere (positive latitudes) yields interesting results (see Fig. 6c and 6d). The last 50 years have actually a higher error than the 50 years before 1900. When calculated over the 20 year average, the correlation coefficient is $\rho = 0.70$, which is significantly lower than for Europe alone ($\rho = 0.85$). Since the areas around the poles are relatively difficult to model, we have tried excluding the polar region in the comparison. This didn't change the results significantly. The same holds when excluding the tropics.

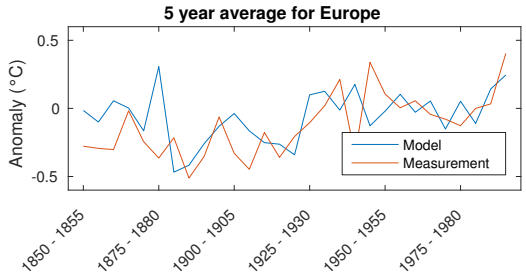
With $RMSE = 0.10°C$, the error is slightly higher than compared to Europe alone.

4.3 Southern Hemisphere

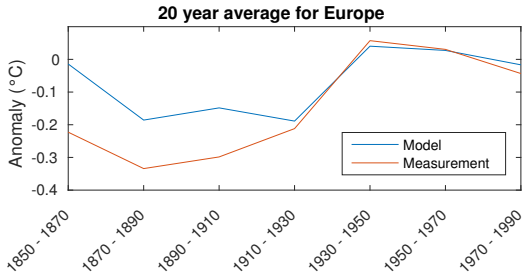
The same observations can be made for the Southern hemisphere (negative latitudes): the RMS error is roughly equal to the RMS error of the Northern hemisphere (Fig. 6e and 6f). The big difference here, is that there is a large error between 1900 and 1930, instead of the last 50 years for the Northern hemisphere.

4.4 Conclusion of space averaged comparison

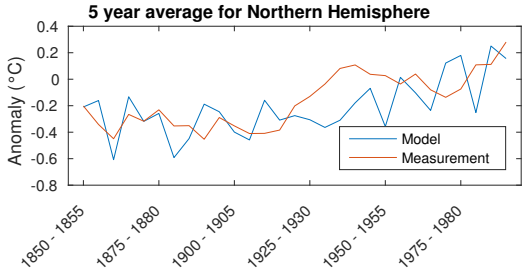
In general, we can conclude that the model output is in better accordance with the data when looking at longer time scales. However, the error between the two strongly depends on the time period we consider, and the area we use. Anomalies for Europe seem to agree well for the last 50 years, whereas there are big differences for this period when looking at the whole Northern Hemisphere. The Southern Hemisphere shows good accordance for the last 50 years, whereas for the years from 1900 to 1950, the model seems to reproduce the anomalies less accurately. However, these conclusions are based upon a rather crude analysis: by simply averaging certain areas and time periods, we loose some of the finer structure and information contained in the datasets. To address this issue, we now develop a method to look at dominant spatial and temporal patterns of the data, using a decomposition into Empirical Orthogonal Functions.



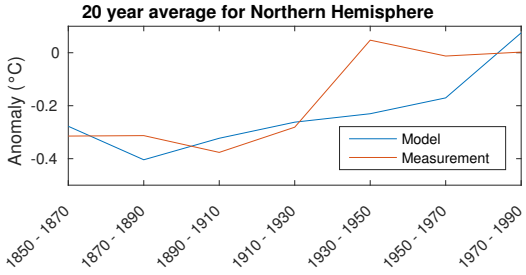
(a) Europe, 5 years
Correlation: $\rho = 0.36$, *RMSE:* $\Delta T = 0.24°C$



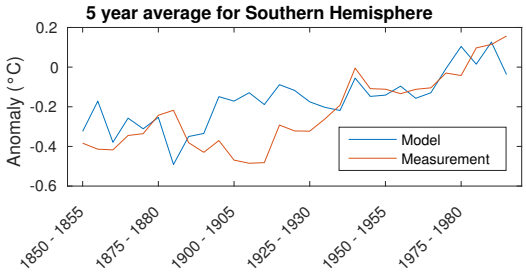
(b) Europe, 20 years
Correlation: $\rho = 0.86$, *RMSE:* $\Delta T = 0.11°C$



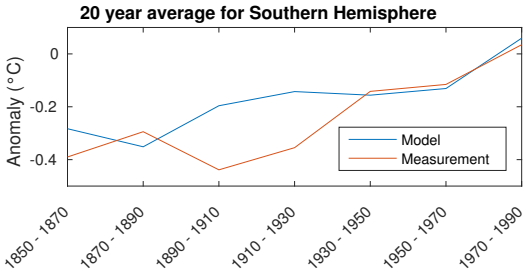
(c) Northern hemisphere, 5 years
Correlation: $\rho = 0.54$, *RMSE:* $\Delta T = 0.21°C$



(d) Northern hemisphere, 20 years
Correlation: $\rho = 0.68$, *RMSE:* $\Delta T = 0.14°C$



(e) Southern hemisphere, 5 years
Correlation: $\rho = 0.65$, *RMSE:* $\Delta T = 0.15°C$



(f) Southern hemisphere, 20 years
Correlation: $\rho = 0.74$, *RMSE:* $\Delta T = 0.12°C$

Figure 6: *Temperature anomalies over Europe (6a, 6b), the Northern (6c, 6d) and the Southern hemisphere (6e, 6f). The left column corresponds to 5 year averages, the right to 20 year averages.*

5 Empirical Orthogonal Functions decomposition

In the previous section, we concluded that agreement between the temperature components of the model and the measurements strongly depends on the area covered, and on the time period we consider. Simply taking averages over a full hemisphere discards information that is actually needed to properly compare the temperature of the model to the dataset.

A better method to analyze variations in space and time is by using Empirical Orthogonal Functions (EOFs). EOF decomposition is used to calculate patterns of highest variability. This allows us to reduce the dimensionality of the system by approximating the dataset by the leading patterns of highest variability.

EOFs have been used in climate science for over 50 years, since their introduction by Obukhov (1947), Lorenz (1956) and Kutzbach (1967). For such an analysis, the dataset is decomposed into a purely time-dependent part, and a purely space-dependent part. Although there is some ambiguity in their respective names, we use the same terminology as Hannachi et al. (2007): we call the time-dependent part *Principal Component* (PC), and the space-dependent part *Empirical Orthogonal Function* (EOF).

We now discuss in section 5.1 how we have to reshape the temperature anomalies, followed by a short overview of how these are decomposed into spatial and temporal components. We then discuss the two most common techniques to perform the EOF decomposition: eigenvalue decomposition in section 5.3 and singular value decomposition in section 5.4. Finally, we apply these techniques in section 5.5.

5.1 Reformatting the data

During this thesis, the data we analyse using EOF decomposition consists of the previously mentioned temperature anomalies on a 64×32 spatial grid, for every month from January 1850 to December 1998 (in total, 1788 months). This has to be put into matrix form first, which is denoted by X. We are using the notation from Björnsson (1997). Every row i of X contains the temperature anomalies at time t_i ($i = 1, \ldots, n$): the 64×32 spatial grid is transformed into a vector $\mathbf{s} = (s_1, \ldots, s_p)$ with $p = 64 \cdot 32 = 2048$ elements.

$$X = \underbrace{\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}}_{\text{Time series for location } j=1} \left.\begin{matrix} \\ \\ \\ \\ \end{matrix}\right\} \begin{matrix} \text{Anomalies for} \\ \text{time } i=1 \end{matrix}$$

Once the data has been shaped into the correct form, it still has to be *centered*. This means that we have to remove the time average from every point. More specifically, this can be split into two steps.

First, for every point in space s_j, we can calculate the average of the time series for the grid point. We write this average $\overline{x}_j$ as:

$$\overline{x}_j = \frac{1}{n} \sum_{i=1}^{n} x_{ij}. \tag{2}$$

By doing this for every $j = 1, \ldots, p$, Eq. (2) can be written using vector notation:

$$\overline{\mathbf{x}} = \frac{1}{n}\, \mathbf{1}_n\, X = \frac{1}{n}\, (1, \ldots, 1)\, X,$$

where $\mathbf{1}_n$ is the row vector with n ones.

Second, the row vector $\overline{\mathbf{x}}$ has to be substracted from every row of X. In other words, the matrix with n copies of the row vector $\overline{\mathbf{x}}$ has to be substracted from X. This can be achieved by multiplying $\overline{\mathbf{x}}$ with the column vector $\mathbf{1}_n^T$, consisting of n ones. The final, centered matrix X', is then equal to:

$$\begin{aligned} X' &= X - \mathbf{1}_n^T\, \overline{\mathbf{x}} = X - \frac{1}{n} \mathbf{1}_n^T \mathbf{1}_n\, X \\ &= \left(I_n - \frac{1}{n} \begin{pmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{pmatrix} \right) X, \end{aligned} \tag{3}$$

where I_n is the $n \times n$ identity matrix.

From now on, we only use X'. Hence, to simplify notation, we drop the prime, and only use X.

5.2 Spatial and temporal decomposition

During an EOF decomposition, we decompose the matrix X into space-dependent modes $u_k(\mathbf{s})$, called EOFs, and time-dependent modes $c_k(t)$, called Principal Components (PC's):

$$X(t, \mathbf{s}) = \sum_{k=1}^{M} u_k(\mathbf{s})\; c_k(t), \tag{4}$$

where M is the number of modes of the system, which is equal to the rank of X. In the next section, we show how to calculate these EOFs and PCs.

5.3 Patterns of highest variability

The interesting part of EOF decomposition comes from the fact that the variables $u_k(\mathbf{s})$ and $c_k(t)$ contain the patterns where variations are largest. This variability is measured using the covariance matrix S, and the patterns are then the eigenvectors of S.

For every two locations i and j, the covariance s_{ij} is defined as:

$$s_{ij} = \frac{1}{n}\sum_{t=1}^{n} x_{ti}x_{tj},$$

since the mean of x is equal to zero. In matrix form, this can be written as:

$$S = \frac{1}{n}X^T X, \tag{5}$$

where X^T is the transpose of X.

Since we try to find a unit-length vector $\mathbf{u} = (u_1, \ldots, u_p)^T$ where the variability of $X\mathbf{u}$ is maximum, we are actually trying to solve the following equation:

$$\max(\mathbf{u}^T S\mathbf{u}), \text{ s.t. } \mathbf{u}^T\mathbf{u} = 1.$$

By the min-max theorem, $\mathbf{u}$ is obtained by finding the eigenvectors of S, so solving this equation:

$$S\mathbf{u} = \lambda^2\mathbf{u}. \tag{6}$$

Since a covariance matrix is positive semidefinite, the eigenvalues of S are positive (Higham, 1988). This is stressed by the notation λ^2.

The k-th eigenvector $\mathbf{u}_k$ is now the k-th EOF (space-dependent pattern). The corresponding k-th PC (time-dependent pattern) can be found by projecting the EOF onto X: $\mathbf{c}_k = X\mathbf{u}_k$.

So far, we have calculated patterns of variability, but we don't know yet which one has the *highest* variability. Let us remind that we have been trying to find a vector $\mathbf{u}$ that maximizes $X\mathbf{u}$. Since $\mathbf{u}_k$ is the k-th eigenvector, corresponding to the k-th eigenvalue λ_k^2, we have:

$$\begin{aligned} S\mathbf{u}_k &= \lambda_k^2\mathbf{u}_k \\ \Leftrightarrow\ \mathbf{u}_k^T S\mathbf{u}_k &= \lambda_k^2\mathbf{u}_k^T\mathbf{u}_k \\ &= \lambda_k^2. \end{aligned} \tag{7}$$

The last step is true since $\mathbf{u}$ is a unit-length vector: $\mathbf{u}^T\mathbf{u} = 1$. Moreover, by plugging in the definition of the covariance matrix from Eq. (5), we find successively:

$$\begin{aligned} \lambda_k^2 &= \mathbf{u}_k^T S\mathbf{u}_k \\ &= \mathbf{u}_k^T \frac{1}{n}X^T X\mathbf{u}_k \\ &= \frac{1}{n}\ \mathbf{u}_k^T X^T\ X\mathbf{u}_k \\ &= \frac{1}{n}\|\mathbf{u}_k^T X^T\|\ \|X\mathbf{u}_k\| \\ &= \frac{1}{n}\|X\mathbf{u}_k\|^2. \end{aligned}$$

This means that the eigenvalue λ_k^2 actually gives us a measure of the variability. This allows us to sort the eigenvalues and corresponding eigenvectors such that $\lambda_1^2 \geq \lambda_2^2 \geq \ldots \geq \lambda_p^2$. Now, the first eigenvector $\mathbf{u}_1$ and corresponding PC $\mathbf{c}_1$ have the highest variability.

Usually, the measure of variability of an EOF/PC pattern is given as a percentage of the total variability. We denote the variability corresponding to the k-th eigenvector by p_k:

$$p_k = \frac{100\ \lambda_k^2}{\sum_{i=1}^{M}\lambda_i^2}\ \%$$

5.4 Calculating EOFs using Singular Value Decomposition

Up until now, to find the patterns of highest variability, we have calculated the eigenvectors of the covariance matrix. While this works perfectly fine, another method to achieve this is by applying Singular Value Decomposition on the anomaly field. This method is computationally much more efficient than solving the set of linear equations of Eq. (6).

In general, Singular Value Decomposition (which we write as SVD from now on), factorizes an $n \times p$ matrix X into three matrices:

- an $n \times n$ orthonormal matrix V ($V^T V = I$, with I the identity matrix),
- a $n \times p$ diagonal matrix Γ, containing the *singular values* on its diagonal
- and a $p \times p$ orthonormal matrix U^T.

Putting these together, we obtain a factorized version of X:

$$X = V\ \Gamma\ U^T \tag{8}$$

These three matrices can be computed using numerical algorithms described in Golub & Reinsch (1970). We now show that V and U are actually the matrices containing the Principal Component vectors (in V) and EOF vectors (in U), which we calculated before using eigenvalue decomposition.

In Eq. (5), we defined the covariance matrix as being $S = \frac{1}{n}X^T X$. By plugging in the SVD of X from Eq. (8), we obtain:

$$\begin{aligned} S &= \frac{1}{n}X^T X \\ &= \frac{1}{n}(V\Gamma U^T)^T(V\Gamma U^T) \\ &= \frac{1}{n}\, U\Gamma^T V^T V\Gamma U^T \\ &= \frac{1}{n}\, U\Gamma^T\Gamma U^T \end{aligned}$$

The last step was possible since V is orthonormal. By rewriting this, we see that this yields the standard form of an eigenvalue decomposition of S:

$$S = U\,(\frac{1}{n}\Gamma^T\Gamma)\,U^T. \tag{9}$$

Note that the eigenvalue decomposition, $S\mathbf{u}_k = \lambda_k^2\mathbf{u}_k = \mathbf{u}_k\lambda_k^2$, can also be written in matrix form. Let U be the matrix where the k-th column is the k-th eigenvector $\mathbf{u}_k$, and let Λ be the diagonal matrix with the eigenvalues λ_k^2 on its diagonal. Then we have:

$$S\,U = U\,\Lambda \tag{10}$$

$$\Leftrightarrow S = U\,\Lambda\,U^{-1} \tag{11}$$

$$\Leftrightarrow S = U\,\Lambda\,U^T. \tag{12}$$

The last step is possible since U is orthonormal.

By comparing Eq. (9) and (12), we see that Singular Value Decomposition of X is equivalent to eigenvalue decomposition of the covariance matrix, with $\Lambda = \frac{1}{n}\Gamma^T\Gamma$.

Just like with eigenvalue decomposition, we can define a measure of variability using the singular values. Let γ_k be the k-th singular value (k-th element on the diagonal of Γ). The amount of variability corresponding to the k-th EOF/PC, p_k, is equal to:

$$p_k = \frac{100\ \frac{1}{n}\gamma_k^2}{\sum_{i=1}^{M}\frac{1}{n}\gamma_i^2}\ \% = \frac{100\ \gamma_k^2}{\sum_{i=1}^{M}\gamma_i^2}\ \%.$$

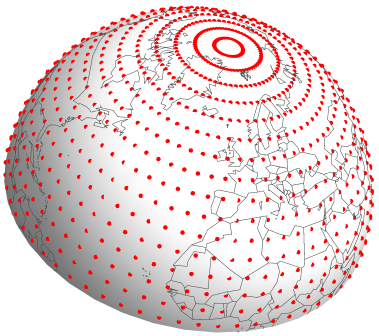


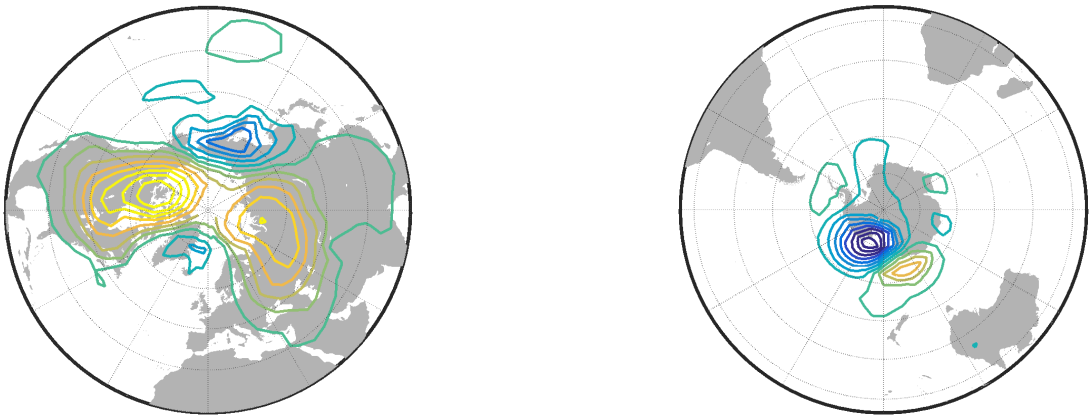
Figure 8: *Distribution of grid points on the Northern Hemisphere. Notice that the distance between the points near the poles is much smaller than near the equator.*

5.5 EOF decomposition of the model state

We apply the above methodology for EOF decomposition on the temperature component of the model. Before we can get started, we have to take into account the fact that there are many more data points near the poles than near the equator. This is because of the projection we use, with constant latitude and longitude differences. The grid points are shown in Fig. 8. To account for the surplus of grid points per unit area near the poles, we have to multiply every point of the temperature component of the model by a weighting factor. This weighting factor should be applied to X^2, which is what we are interested in (because $S = \frac{1}{n}X^T X$). The area of a spherical cap is proportional to the cosine of the latitude. Therefore, the weighting factor of X^2 is then the cosine of the latitude, which is equivalent to applying a weighting factor of the square root of the cosine of the latitude to the original matrix X. This is discussed in more detail in Baldwin et al. (2009).

Once the weighting is performed, we can proceed to the actual EOF decomposition. First, we reshaped the $1788 \times 64 \times 32$ matrix from section 4 (1788 time moments on a 64×32 grid), into a $n \times p$ matrix, with $n = 1788$ and $p = 64 * 32 = 2048$. Then we centered it by using Eq. (3), to obtain the anomaly field X. Moreover, we calculated the covariance matrix S as in Eq. (5).

We obtained the EOFs by calculating the eigenvectors of S by solving Eq. (6). Finally, the corresponding PCs were calculated as $\mathbf{c}_k = X\mathbf{u}_k$. The first EOF of the



(a) *First EOF of the model, calculated over the Northern hemisphere (15.9% of the variability)*

(b) *First EOF of the model, calculated over the Southern hemisphere (15.6% of the variability)*

Figure 7: *First EOFs of the model. When calculating the EOFs over the whole Earth, the variabilities of the Northern hemisphere are much bigger than the Southern hemisphere. For this reason, it makes more sense to split the calculations over the Northern and Southern hemisphere.*

Northern and of the Southern hemisphere are shown in Fig. 7.

6 EOF decomposition with missing values

We have seen that EOF decomposition is a powerful tool to determine the patterns of highest variability, both in time (PC) and in space (EOF). If we want to apply this to the data (the measurements from the HadCRUT dataset), we are faced with a big problem: a lot of datapoints are missing. This means that we cannot calculate a covariance matrix, or perform SVD decomposition.

There are various methods to overcome this problem. We have chosen for the *Data Interpolating Empirical Orthogonal Functions* (DINEOF) method, developed by Beckers & Rixen (2003). It starts with an initial guess for the missing values of all zeros, and performs the EOF decomposition. The field is then reconstructed using a truncated set of EOFs, and the missing values are replaced by the values in this reconstructed field. This is repeated until convergence.

DINEOF is found to perform better at reconstructing datasets than the two other most commonly used methods: *Least-squares estimation of coefficients* (LSEOF) and *Recursively-Substracted EOFs* (RSEOF), as discussed in Taylor et al. (2013). It is, however, much more computationally expensive.

In section 6.1, we discuss in more detail the mathematics of DINEOF, followed by a method to determine the stopping criteria. Since DINEOF reconstructs the field using a truncated set of EOFs, we need a way to determine the optimal number of EOFs for this truncation. This is described in section 6.2. Finally, we apply DINEOF to the data in section 6.3.

6.1 Data Interpolating EOF

The most trivial way to handle missing values is by replacing them by their unbiased estimator, which is 0 for a set of anomalies. This is not very accurate, though.

The spatial and temporal patterns found with EOF decomposition represent the large scale structures of the data. These patterns take into account the whole dataset. It is therefore logical to use these patterns to improve the initial guess.

Since only the patterns of highest variability can really say something about the value of the missing datapoints, it is important to reconstruct the field using a *truncated* set of EOFs/PCs. We will see in section 6.2 how to determine this optimal number N^* of EOFs.

Let X_0 be the $n \times p$ matrix with measurements, as

defined in section 5.1, where every missing element is replaced by 0. We gradually change the missing values with better estimates, until we reach the reconstructed field X_r. Let ${}^iV\ {}^i\Gamma\ ({}^iU)^T$ be the singular value decomposition of the matrix X_i, as discussed in section 5.4. The DINEOF algorithm now consists of these steps:

1. Start with the initial guess X_0.
2. Perform the SVD decomposition of X_0 to obtain
$$V\ \Gamma\ U^T \overset{\text{SVD}}{=} X_0.$$
3. Let $X_{R,i}$ be the reconstructed field at step i using only the N first EOFs/PCs:
$$X_{R,0} = V_{N^*}\ \Gamma_{N^*}\ (U_{N^*})^T,$$
where V_{N^*} is the matrix consisting of only the first N^* columns (similarly for U_{N^*}).
4. For every missing datapoint, replace the initial guess by the value of the reconstructed field $X_{R,0}$:
$$X_1 = X_0 + \delta(X_{R,0}).$$
Here, $\delta(\cdot)$ is a function which leaves any element of a matrix untouched if it was a missing value, or replaces the element by 0 if it was not a missing value.
5. Repeat step 2, 3 and 4 until convergence:
$$\begin{cases} V\ \Gamma\ U^T \overset{\text{SVD}}{=} X_i \\ X_{R,i} = V_{N^*}\ \Gamma_{N^*}\ (U_{N^*})^T \\ X_{i+1} = X_0 + \delta(X_{R,i+1}) \end{cases}$$

The stopping criteria is then reached when the relative difference between two consecutive steps is below a certain threshold τ, in other words, when:
$$\frac{\|X_{i+1} - X_i\|_1}{\|X_{i+1}\|_1} \leq \tau.$$
Here, $\|X\|_1$ is the *matrix 1-norm* of an $n \times p$ matrix X, defined by:
$$\|X\|_1 = \sum_{i=1}^{n}\sum_{j=1}^{p} \left|X_{ij}\right|.$$
Typically, $\tau = 0.02$, but any small value could be taken.

We now show a method to determine the optimal number N^* of EOFs to be used in the truncation.

6.2 Optimal number of EOFs for truncation

As discussed in section 6.1, the DINEOF method depends on truncation of the number of EOFs. In order to assess the quality of an EOF interpolation using a finite number of modes, we need to define a reference, and an error to compare the reference to the results of DINEOF interpolation.

Therefore, we apply the DINEOF algorithm to two datasets: first, the original set of measurements X, and second, the same set X, where we treat a number q of existing values as missing. This second set is called X_{test}. Let $J = \{j_1, .., j_q\}$ be the set of the indices of the q values treated as missing in X_{test}, taken at random. Typically, q is a small fraction of the number of elements of X. During this thesis, we use $q = 0.1\%(n \cdot p) \approx 3500$.

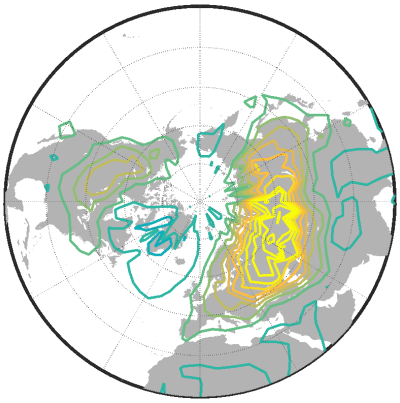
After applying the EOF interpolation, we have to compare the resulting matrices X and X_{test}. To do this, we define the error as the root mean square error (RMSE) between the two sets of values of X and X_{test} with indices in J. The RMSE is defined by:
$$RMSE := \sqrt{\frac{1}{q}\sum_{j\in J}\left(X_{test,j} - X_j\right)^2}$$

We now perform the DINEOF algorithm repeatedly with different truncation for the number of EOFs, and calculate the RMSE. The number of EOFs corresponding to the lowest RMSE is then the optimal number N^*.

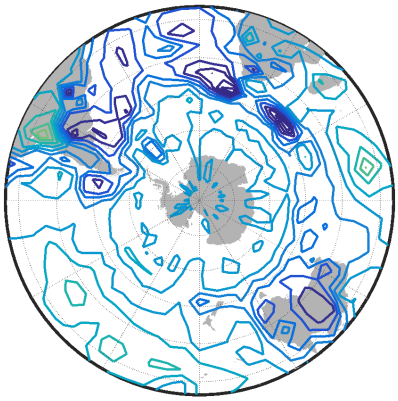
6.3 DINEOF decomposition applied to the data

The goal of the DINEOF method in this thesis is to obtain an EOF decomposition of the HadCRUT dataset. As mentioned before, this dataset doesn't provide temperature measurements for every grid point over the full period 1850-1998. The coverage, defined as the percentage of grid points with existing values (not `nan`) per time moment, is plotted in Fig. 2.

DINEOF decomposition provides a method to calculate EOFs even with this partial coverage. We first have to reformat the HadCRUT dataset to a $n \times p$ matrix ($n = 1788$ time steps, $p = 64 * 32 = 2048$ grid points), as described in section 5.1. Once this is done, we still need to determine N^*, the optimal number of



(a) *First EOF of the HadCRUT dataset, calculated over the Northern hemisphere (11.1% of the variability)*



(b) *First EOF of the HadCRUT dataset, calculated over the Southern hemisphere (9.5% of the variability)*

Figure 9: *First EOFs of the HadCRUT dataset, calculated using the DINEOF algorithm. When calculating the EOFs over the whole Earth, the variabilities of the Northern hemisphere are much bigger than the Southern hemisphere. For this reason, it makes more sense to split the calculations over the Northern and Southern hemisphere.*

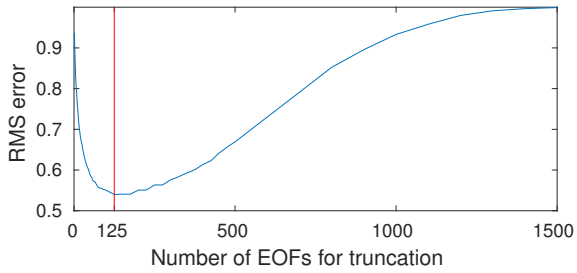


Figure 10: *Root mean square error between the set of values treated as missing and the reference set, as a function of the number of EOFs for DINEOF truncation. For the HadCRUT dataset, this yields a minimum at* $N = 125$.

EOFs for the truncation steps. To do this, we start by selecting a small number of existing measurements. We used a random sample of 0.1% of the existing values, which is approximately $q = 3500$ data points. We then start the DINEOF algorithm with 1 mode, and measure the RMS error with the reference X_{test}. This is repeated for every number of EOFs until 1788, which is equal to taking all EOFs. The root mean square error as a function of the number of EOFs for truncation is plotted in Fig. 10. For the HadCRUT dataset, $N^* = 125$.

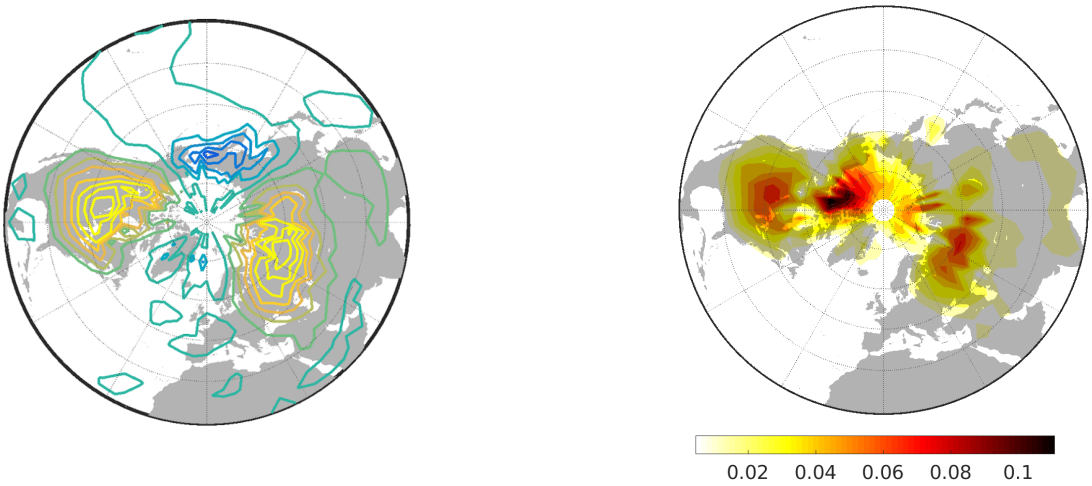
With the newly calculated optimal number of EOFs N, we can perform the DINEOF algorithm on the HadCRUT dataset. Just like with the EOFs of the temperature component of the model, the Northern hemisphere has much more variability than the Southern hemisphere. It makes therefore more sense to look at the two hemispheres seperately. The first EOF for both the North and South pole are shown in Fig. 9.

The next step is now to look at how good this DINEOF decomposition actually is.

6.4 DINEOF validation

Since the HadCRUT only has a partial coverage, we have no way to easily determine the accuracy of the reconstructed EOFs. We do, however, have a full coverage with the model, with corresponding exact EOFs (section 5.5). To test the DINEOF method applied to the HadCRUT data, we use the model temperature component, and treat a number of values as non existing. These values are exactly the missing values in the HadCRUT dataset. This gives an incomplete dataset similar in coverage to the HadCRUT dataset, but now we know what these missing values should be.

From this dataset, we calculate the EOFs using the DINEOF method. These are then compared to the actual EOFs. The first reconstructed EOF of the Northern hemisphere, along with the absolute difference between the reconstructed and original EOF, defined



(a) *First reconstructed EOF of the model temperature component, with a number of missing values. These correspond to the missing values of the HadCRUT dataset. Note the resemblence to the original EOF, Fig. 7a.*

(b) *Absolute difference, in °C, between the original first EOF (Fig. 7a) of the model, and the reconstructed EOF (Fig. 11a).*

Figure 11: *Validation of the DINEOF method using a subset of the values of the output of the model. Everywhere except around the pole, the reconstruction is good: the same large scale patterns as in the original EOF are visible. The lack of data points around the poles make the reconstruction very difficult in these areas.*

as:

$$\text{abs.diff.} = \left|\text{EOF}_{reconstr} - \text{EOF}_{orig}\right|,$$

is shown in Fig. 11. From this figure, we can see that the reconstruction is good for most of the Northern Hemisphere. Around the poles, however, the reconstruction is seriously influenced by the lack of measurements.

More quantitatively, the correlation coefficient between the original EOF and the reconstructed EOF is $\rho = 0.65$, which indicates the reconstruction is good on average. From this, we can conclude that thanks to the DINEOF reconstruction, we have a working method to calculate EOF/PCs of the HadCRUT dataset with partial coverage.

7 EOF based comparison of the model to the data

When calculating EOF/PCs of a dataset, it is decomposed into spatial patterns (EOFs) and corresponding time-dependent parts (principal components, PCs). In section 5 and 6, we discussed a method to obtain an EOF/PC decomposition of the model temperature component and of the HadCRUT dataset. Since we want to know how the model compares to the measurements, we are interested in comparing their EOF/PCs. However, we can't simply compare the first EOF of the model with the first EOF of the measurements, since they are calculated with respect to a different time-dependent part, a different principal component.

In this section, we discuss a method to still be able to compare the respective EOF/PCs. First we shortly develop the theoretical background in section 7.1, and continue with the results when applied to the model and measurements in section 7.2. In that section, we also look at what happens when the datasets are averaged over 1, 2, 5, 10 and 20 years respectively.

7.1 Projections

We have already seen in Eq. (4) that a dataset X can be decomposed in a spatial component $u_k(\mathbf{s})$ and a time component $c_k(t)$:

$$X(t, \mathbf{s}) = \sum_{k=1}^{M} u_k(\mathbf{s})\, c_k(t),$$

where M is the total number of EOF/PCs. This can be calculated using Singular Value Decomposition:

$$\begin{aligned} X(t, \mathbf{s}) &= A\ \Gamma\ U^T \\ &:= C\ U^T, \end{aligned}$$

with $C := A\ \Gamma$. Now, the matrix C contains the principal components as its column vectors, and U contains the EOFs, also as its column vectors. This way, we obtain a PC and EOF matrix for the model, and for the HadCRUT dataset:

$$\begin{cases} X_{\text{model}} = C_{\text{model}}\ U^T_{\text{model}} \\ X_{\text{data}} = C_{\text{data}}\ U^T_{\text{data}}. \end{cases}$$

Both X_{model} and X_{data} depend upon a spatial component and a time component. This makes it difficult to compare, for example, the first EOF of the model with the first EOF of the data, since they correspond to different time components. The trick to still be able to compare the two, is to use either a common spatial pattern, or a common PC, between the model and the data. We are keeping the spatial pattern fixed, and use the model's EOFs as common spatial component. We now have to find a new PC $\widehat{C}_{\text{data}}$ for the HadCRUT dataset, such that

$$\begin{cases} X_{\text{model}} = C_{\text{model}}\ U^T_{\text{model}} \\ X_{\text{data}} = \widehat{C}_{\text{data}}\ U^T_{\text{model}}. \end{cases}$$

Since the EOF matrix U is orthonormal (from the definition of Singular Value Decomposition), we know that $U^T\ U = I$, with I the identity matrix. With this, we find successively:

$$\begin{aligned} & \widehat{C}_{\text{data}}\ U^T_{\text{model}} = X_{\text{data}} \\ \Leftrightarrow\ & \widehat{C}_{\text{data}}\ U^T_{\text{model}}\ U_{\text{model}} = X_{\text{data}}\ U_{\text{model}} \\ \Leftrightarrow\ & \widehat{C}_{\text{data}} = X_{\text{data}}\ U_{\text{model}}. \end{aligned}$$

This also explains the name of this method: *projection analysis*, since we are projecting one dataset onto the spatial patterns of the other dataset (the model).

It now becomes possible to properly compare the model and the dataset by comparing C_{model} and $\widehat{C}_{\text{data}}$.

Similarly, we can project the model dataset onto the HadCRUT EOFs. We then obtain $\widehat{C}_{\text{model}} := X_{\text{model}}\ U_{\text{data}}$, which can be compared to C_{data}. We do this by looking at the correlation coefficient between $\widehat{C}_{\text{model}}$ and C_{data}, defined as:

$$\rho(C1, C2) = \frac{1}{n-1} \sum_{i=1}^{n} \left(\frac{C1_i - \mu_{C1}}{\sigma_{C1}} \right) \left(\frac{C2_i - \mu_{C2}}{\sigma_{C2}} \right),$$

with μ_C the mean of the principal component C, and σ_C its standard deviation: $\sigma_C = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (C_i - \mu_C)^2}$.

7.2 Results

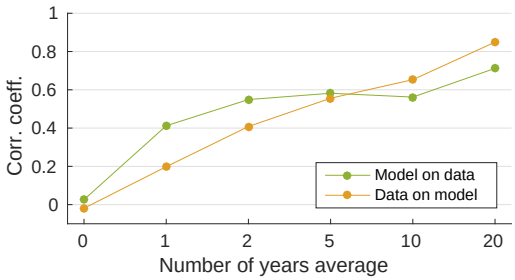
As explained in the previous section, by projecting the model temperature component onto the EOFs of the data (which we'll call *model on data*), or the HadCRUT data onto the EOFs of the model (*data on model*), we obtain a new principal component, which we can compare to the PC of the data or the model, respectively. Just like with the spatial averaging (without EOFs) of the beginning, we use two measures of skill: one is given by the correlation coefficient between the two PCs, the other by the root mean square error between the two. Moreover, we consider the Northern and the Southern hemisphere separately.

When using the monthly data, we find very low correlation coefficients: for the Northern Hemisphere, 0.026 for the *model on data*, and -0.019 for the *data on model* projection. There is practically no correlation between the two PCs. Moreover, the RMS errors are large as well. This is because the LOVECLIM model is designed for long term climate trends. The model is clearly not accurate on the small time scales.

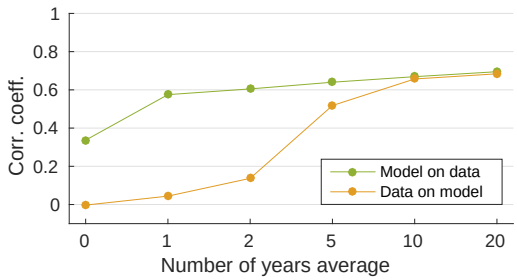
It is therefore only natural to perform the projection analysis after averaging over the time. When we average the temperature component of the model and the HadCRUT dataset over a period of 1, 2, 5, 10 and 20 years respectively, the correlation coefficients become much larger, and the RMS errors decrease. This is shown in Fig. 12. We also include the full principal components for the Northern Hemisphere averaged over 5 years (Fig. 13a for the data on model, Fig. 13c for the model on data) and 20 years (Fig. 13b for the data on model, Fig. 13d for the model on data).

7.2.1 Northern Hemisphere

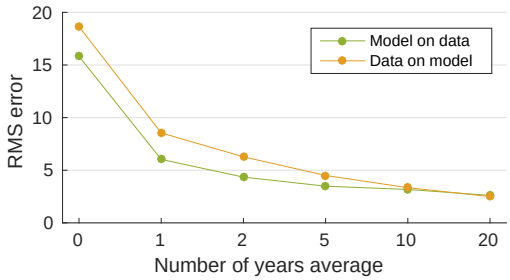
The correlation coefficients and RMS errors of the Northern Hemisphere (Fig. 12a) show two main results. First of all, the smaller the time scale we look at, the lower the correlation and the higher the RMS error. This confirms that the model is in better accordance with the data on longer time scales. Secondly, the *model on data* projection has, in general, a higher correlation and lower RMS error than the *data on model* projection. This can be explained by the fact that the



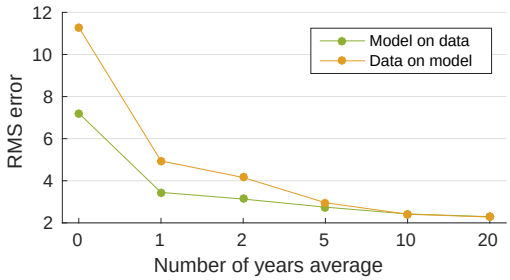
(a) *Correlation, Northern hemisphere*



(b) *Correlation, Southern hemisphere*



(c) *RMS error, Northern hemisphere*



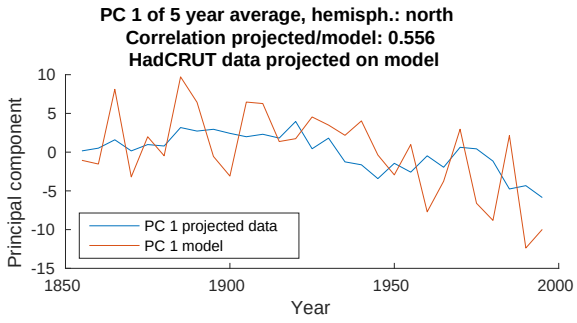
(d) *RMS error, Southern hemisphere*

Figure 12: Correlation coefficients. *First the data is averaged over 0 (monthly), 1, 2, 5, 10 and 20 years respectively. Then, the correlation between a principal component and the other projected principal component are calculated. In green, between the first PC of the HadCRUT data (C_{data}) and the projected PC of the model onto the EOFs of the data ($\widehat{C}_{model} = X_{model}\ U_{data}$). In orange, between the first PC of the model (C_{model}) and the projected PC of the data onto the EOFs of the model ($\widehat{C}_{data} = X_{data}\ U_{model}$). Fig. (c) and (d) represent the root mean square error between the two PCs.*

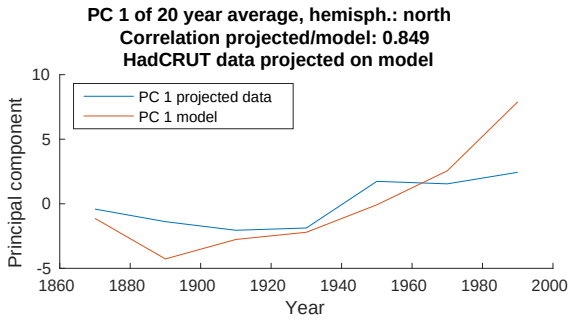
data on model projection keeps the spatial patterns of the model fixed, and finds a corresponding principal component of the data. The first spatial pattern (EOF) of the model shows strong variability around Northern Canada and Alaska (see Fig. 7a). These are regions where the HadCRUT dataset has less observations than farther away from the poles, as can be seen in Fig. 1a. The data is therefore less accurate in these regions, which leads to lower correlation coefficients. However, when considering longer time scales, (≥ 5 year averages), the *data on model* and *model on data* projection yield correlations of approximately the same magnitude.

7.2.2 Southern Hemisphere

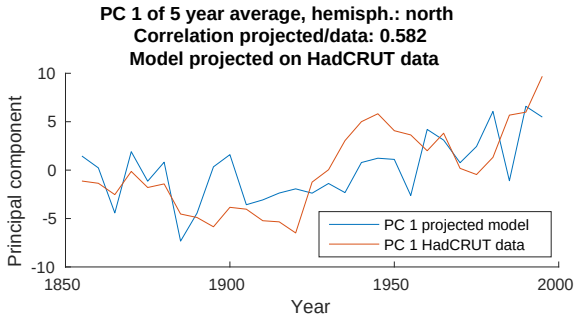
The correlation coefficients of the Southern Hemisphere (Fig. 12b) show similar features to the Northern Hemisphere. Small time scales still have lower correlation coefficients than longer time scales, and *model on data* yields higher correlation than the *data on model*. The difference between the two projection modes is much more striking for the Southern Hemisphere than for the Northern Hemisphere, especially for time scales between monthly and 2 years. This would suggest that the model performs well at these time scales. However, when looking at the RMS errors, these are high as well, especially for monthly averages. Moreover, the Southern Hemisphere contains lots of oceans, with a lower coverage in the HadCRUT dataset. By projecting the model anomalies on the EOF of the data, we are projecting onto an EOF that was created using lots of missing values. We therefore cannot draw striking conclusions on the quality of the model using these correlation coefficients. We would need more complete data for this.



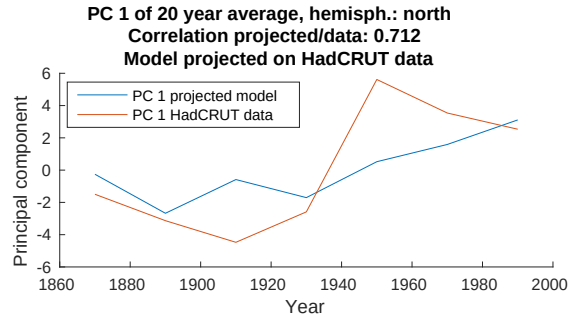
(a)



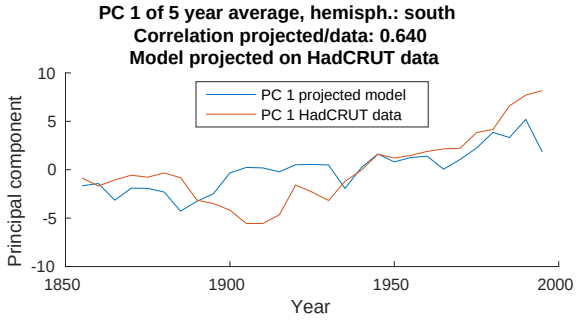
(b)



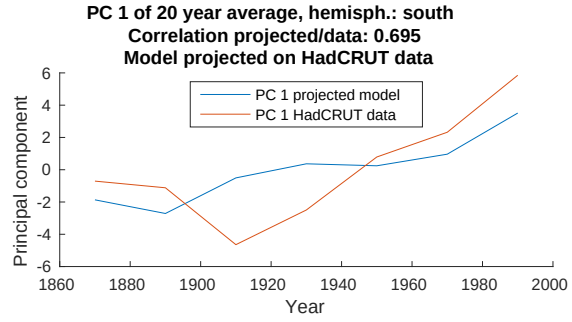
(c)



(d)



(e)



(f)

Figure 13: First principal component *of the model and the HadCRUT dataset. For figure (a) and (b), the HadCRUT dataset is projected onto the model, for figures (c), (d), (e) and (f), the model is projected onto the data. This is shown on the left for 5 year average, on the right for 20 year average.*

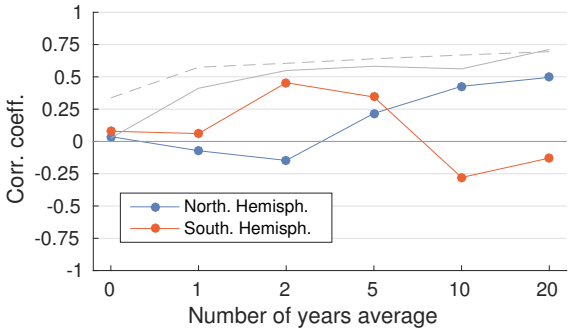


Figure 14: *Correlation coefficients between the second projected PC of the model ($\widehat{C}_{model,2}$) and the second PC of the data ($C_{data,2}$). The solid and dashed gray lines are the reference corr.coeff. of the first PCs (correlation between $\widehat{C}_{model,1}$ and $C_{data,1}$) for the Northern and Southern Hemisphere, respectively.*

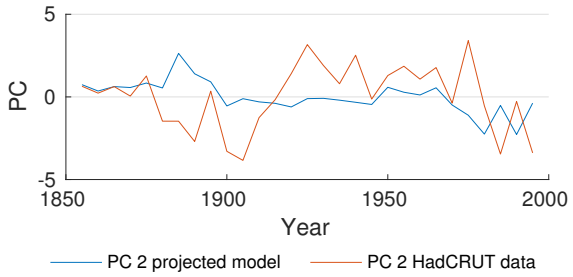


Figure 15: *Projection analysis with the* second *principal components for the Southern Hemisphere, 5 year averages. The values of the projected PC are very close to zero.*

7.2.3 Second Principal Component

Up until now, we have only considered the projection of the first PC onto the first EOF. These are the most important EOF/PCs, since they represent the dominant patterns of variability. We can also look at the projection using the second PC/EOFs. These however lead to very low correlation coefficients. In Fig. 14, the correlation coefficients of the *model on data* projections using the second EOF/PCs can be seen. For time scales up to 5 years, the correlations are smaller than 0.25. For the Southern Hemisphere, the correlation even becomes negative.

The explanation for this is not straightforward. These second EOF/PCs represent patterns of lesser variability than the first EOF/PCs. Moreover, correlation alone doesn't tell everything about the analysis. The projected second PC of the model on the data is much smaller in absolute value than the second PC of the data (see Fig. 15, where the projected PC is 3 times smaller on average than the data PC).

This means that the model hardly has any variation along the spatial pattern of the second EOF of the data, which makes the model more difficult to compare to the data. For this reason, we only consider the first EOF/PC.

7.3 Conclusion of EOF projection analysis

The projection analysis gives a quantitative measure of how good the model anomalies compare to the HadCRUT dataset, by considering the patterns of highest variability. This method confirms the idea that the LOVECLIM model performs poorly on small time scales, like monthly, and much better for longer time scales.

For the Northern Hemisphere, correlations between the first principal component of the data and model are almost zero, and the RMS errors very big. Moreover, *model on data* correlations are generally higher and RMS errors lower than for the *data on model* projections. This is can be explained by the fact that the first EOF of the model is centered around polar regions, where the HadCRUT misses a lot of measurements. For longer time scales ($\geq$ 5 year averages), the model anomalies are in rather good accordance with the HadCRUT dataset, with correlation coefficients larger than 0.5, and small RMS errors.

The Southern Hemisphere is slightly more complicated. When projecting the data on the EOF of the model reconstruction, the correlation coefficients follow a similar trend as for the Northern Hemisphere: almost 0 for monthly averages, very low for time scales of 2 years or smaller, and quite high for 5 years or more. Projecting the model anomalies on the data EOF yields very high correlations, even for monthly averages. However, the RMS error are still very large. The high correlations can be explained by the fact that the Southern Hemisphere is made of mostly oceans,

where historical accurate measurements are scarce. Projecting the model anomalies on the EOF pattern of the data actually discards much information of the model, since large parts of the data EOF are zero or very small because of the missing values. We are then actually comparing the model to the data on a very limited region. For this reason, we cannot draw conclusions on the quality of the model on the Southern Hemisphere using this analysis.

We have shown an extensive method to compare the LOVECLIM model to the HadCRUT dataset. Instead of simply comparing, the next section gives a method to *improve* the model output using another set of measurements. With this method, called *data assimilation*, the goal is to obtain better accordance with a reference dataset (in our case, the HadCRUT dataset). A second goal is to give an estimate of the variance of the model, which we assume to be unknown. This is treated in section 8.8.

8 Data assimilation

Until now, we have provided a methodology to compare a model output to a dataset of measurements. This can be done in the most rudimentary form by simply averaging over certain regions in space and in time for both the model and the data, then comparing the two (section 4). A more refined method, which takes into account the regions with greatest variability and incomplete datasets, is obtained by decomposing the datasets into spatial and temporal components using EOF decomposition, and comparing two principal components by projecting one dataset onto the EOFs of the other dataset.

Once we know how well a model compares to the measurements, we can try to improve the model by incorporating certain measurements in the model. This is called *data assimilation*.

8.1 A brief history of data assimilation

Early implementations of data assimilation date back to the 1950's, when the first numerical computer aided models appeared. It was called *objective analysis*, aimed at modernising a time where only *subjective analysis* was used: weather scientists used graphical tools, drawings and their own interpretation to use the available measurements to predict the weather. Objective analysis was already applied by Gilchrist & Cressman in 1954 and Bergthórsson & Döös in 1955. This relied on polynomial interpolation of the available measurements, which were considered as exact.

Gradually, weather and climate scientists began to use two facts about models and measurements. First, a model is but an approximation of reality: it uses simplified (geo)physical laws of motion and has a finite numerical precision. However, a model is in principle available for every point in space and in time. On the other hand, measurements are typically sparse, incomplete and also contain errors. This lead to the advancement of *statistical* methods for data assimilation. These used a covariance for the model, and for the measurements. This method is called *optimal interpolation*, and first used by Norwegian meteorologist Eliassen (1954) and Soviet mathematician Gandin (1963). The basic idea of this method is to obtain a new analyzed field, which combines both the model (called "background field") and the measurements ("observation field"). The weights of these two fields are chosen such that the error in the analyzed field (difference between analyzed field and true field) is minimum.

The next idea in data assimilation was, instead of combining the model output values with the measurements, now the measurements would be used to obtain a better initial condition of the model. For example, today's temperature measurements would be used to calculate an initial condition for the model for yesterday. These methods are called *variational data assimilation*. Some methods include only the current measurements. These methods are called 3D-Var, explained in more detail in Courtier et al. (1998). Other methods include the current and past measurements. Because of the added time dimension, these methods are called 4D-Var, like described by Thepaut & Courtier (1991).

There exist many other methods of data assimilation, like ensemble data assimilation, which performs data assimilation for a whole set, called *ensemble*, of initial conditions. Here, we focus on the so called Kalman filter.

8.2 Offline data assimilation

Let's go back to the content of this thesis. We have been comparing some output of a climate model with a certain dataset of measurements. The next step is now to obtain an improved set of temperature data by combining the model and the measurement. Since we only use the output of the model, all the newer types of data assimilation become unusable: we cannot change the model's initial conditions, and we cannot rerun the model with different parameters. For weather models, this is not a viable assumption, since we need updated weather conditions every day. However, the LOVECLIM model is a long term paleoclimate model (simulating the model in the past), and a new run takes up very much computing power.

A cheap and fast alternative to improve the output of the model is to use *offline* data assimilation. This only alters the output itself, in contrast to *online* data assimilation, which requires new runs and changes to the model. At every timestep t, we combine the background field (model output of the whole Earth at time t) with the observation field (measurements available of time t). This is comparable to performing, at every timestep, a weighted average of the model and the measurements.

In the next section, we discuss the theory behind the Optimal Interpolation data assimilation method we use, which we call *Kalman filter data assimilation*, named after the Hungarian American engineer Rudolph Kalman who first developed this methodology (Kalman, 1960).

8.3 Kalman filter data assimilation

The data assimilation method we use in this thesis is discussed in extensive detail in Ghil (1989). We elaborate on the most important results.

8.3.1 Kalman filter for a scalar

Combining a background field with an observation field requires a lot of matrix manipulation, which makes the discussion about this method more complicated. To better understand the method, we start with a simplified version, where our background field is a single variable y, and the observation field a single variable z. These are both estimates of the true field, the variable x. The goal is to find the best estimate $\hat{x}$ of x.

We state that our estimate $\hat{x}$ is some linear combination of y and z:

$$\hat{x} = \alpha y + \beta z, \tag{13}$$

where α and β are the respective weights of y and z. The first assumption we make is that y and z are unbiased estimates of x, meaning that their expectation values are equal: $\mathrm{E}[y] = \mathrm{E}[z] = \mathrm{E}[\hat{x}] = \mathrm{E}[x]$. From this, we can show that $\alpha + \beta = 1$:

$$\begin{aligned}\mathrm{E}[x] &= \mathrm{E}[\hat{x}] \\ &= \mathrm{E}[\alpha y + \beta z] \\ &= \alpha\,\mathrm{E}[y] + \beta\,\mathrm{E}[z] \\ &= \alpha\,\mathrm{E}[x] + \beta\,\mathrm{E}[x] \\ &= (\alpha + \beta)\,\mathrm{E}[x].\end{aligned}$$

Since $\mathrm{E}[x] = (\alpha + \beta)\,\mathrm{E}[x]$, we can indeed conclude that $\alpha + \beta = 1$. We can now rewrite Eq. (13) to:

$$\hat{x} = y + \beta(z - y).$$

The second assumption concerns the errors of y and z. We assume that y has (known) variance $\sigma_1^2 := \mathrm{E}[y - x]^2$, just like z has (known) variance $\sigma_2^2 := \mathrm{E}[z - x]^2$. Moreover, we assume that those errors are uncorrelated:

$$\mathrm{E}[(y - x)(z - x)] = 0. \tag{14}$$

The goal is to find the weights α and β that will minimize the variance of the estimate $\hat{x}$. We call this variance σ^2. Following these steps successively, we find that:

$$\begin{aligned}\sigma^2 &:= \mathrm{E}[\hat{x} - x]^2 \\ &= \mathrm{E}[\hat{x} - (\alpha + \beta)x]^2 \\ &\qquad \text{(because } \alpha + \beta = 1\text{)} \\ &= \mathrm{E}[\alpha y + \beta z - \alpha x - \beta x]^2 \\ &= \mathrm{E}[\alpha(y - x) + \beta(z - x)]^2 \\ &= \alpha^2\,\mathrm{E}[y - x]^2 + \beta^2\,\mathrm{E}[z - x]^2 \\ &\qquad + 2\alpha\beta\,\mathrm{E}[(y - x)(z - x)].\end{aligned}$$

We can further simplify this formula by plugging in the variance of y and z, and by using the fact that the

errors in y and z are uncorrelated (Eq. (14)). We then obtain:

$$\begin{aligned}\sigma^2 &= \alpha^2\sigma_1^2 + \beta^2\sigma_2^2 \\ &= (1-\beta)^2\sigma_1^2 + \beta^2\sigma_2^2.\end{aligned}$$

By taking the derivative, and setting it to zero, we obtain the equation for the optimal weight β^*:

$$\begin{aligned}-2(1-\beta^*)\sigma_1^2 + 2\beta^*\sigma_2^2 &= 0 \\ \Leftrightarrow \quad \beta^* &= \frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}.\end{aligned}$$

We now have an optimal expression for our original problem $\hat{x} = y + \beta(z-y)$, with $\beta = \beta^*$ the optimal weight of the difference $(z-y)$. The factor β^* is also called the *gain factor*.

The next step is to generalize this for the full background field and observation field.

8.3.2 Kalman filter for space and time-dependent fields

Just like in the simplified case, we want to find an estimate of a true value, using two sources: the background field, and the observation field. The quality of this estimated field is quantified by looking at the analyzed error covariance matrix. This time, we consider a vector background field $\mathbf{w}_k^b$ and a vector observation field $\mathbf{w}_k^o$ at time k. These fields can contain several physical quantities: temperature, pressure, humidity, etc. We use these fields to produce an analyzed field, $\mathbf{w}_k^a$. Similarly to the simplified case, where we had $\hat{x} = y + \beta(z-y)$, we now have:

$$\mathbf{w}_k^a = \mathbf{w}_k^b + \mathbf{K}_k(\mathbf{w}_k^o - \mathbf{w}_k^b), \tag{15}$$

where $\mathbf{K}_k$, called the *Kalman gain*, takes the role of the weight β.

The problem with this formulation, is that the number of elements of $\mathbf{w}_k^o$ should be equal to the number of elements of $\mathbf{w}_k^b$, and that those elements should represent the same quantity at the same locations. Typically this is not the case: the number of observations is usually much lower than the number of elements in the background field (the model output). Moreover, observations can be measurements of a different physical quantity: satellites can often only measure radiance in the atmosphere, and not temperature or humidity which are used in a model (Eyre, 1989). Mathematically, $\mathbf{w}_k^b$ and $\mathbf{w}_k^o$ reside in different vector spaces, with usually $\dim(\mathbf{w}_k^o) \ll \dim(\mathbf{w}_k^b)$.

We account for this by introducing a so called *observation operator* $\mathbf{H}_k$, which can be different at every timestep. This operator maps the background field to the phase space of the observation field. Practically speaking, if both $\mathbf{w}_k^b$ and $\mathbf{w}_k^o$ measure the same physical quantity, $\mathbf{H}_k$ maps $\mathbf{w}_k^b$ to the locations of the observations. In our case, $\mathbf{w}_k^b$ is an element of $\mathbb{R}^p$ and $\mathbf{w}_k^o$ an element of $\mathbb{R}^{\dim(\mathbf{w}_k^o)}$. Eq. (15) becomes:

$$\mathbf{w}_k^a = \mathbf{w}_k^b + \mathbf{K}_k(\mathbf{w}_k^o - \mathbf{H}_k\mathbf{w}_k^b). \tag{16}$$

The next step is to take into account the statistics of the system. Let's introduce the *true* field, $\mathbf{w}_k^t$. This is similar to x in the simplified example. We first consider the observations. We assume that the observations come from the true field, with some additional noise term $\mathbf{b}_k^o$. Mathematically, we can represent this assumption as:

$$\mathbf{w}_k^o = \mathbf{H}_k\mathbf{w}_k^t + \mathbf{b}_k^o. \tag{17}$$

The noise term $\mathbf{b}_k^o$ is considered to be white noise: $\mathrm{E}[\mathbf{b}_k^o] = 0$ and $\mathrm{E}[\mathbf{b}_k^o(\mathbf{b}_k^o)^T] = \mathbf{R}_k$. This last term $\mathbf{R}_k$ is called the *observation error covariance matrix*.

Likewise, the background term (corresponding to the model) also has an error covariance matrix, defined by:

$$\mathbf{P}_k^b := \mathrm{E}[(\mathbf{w}_k^b - \mathbf{w}_k^t)(\mathbf{w}_k^b - \mathbf{w}_k^t)^T], \tag{18}$$

and the analyzed term has an error covariance matrix:

$$\mathbf{P}_k^a := \mathrm{E}[(\mathbf{w}_k^a - \mathbf{w}_k^t)(\mathbf{w}_k^a - \mathbf{w}_k^t)^T]. \tag{19}$$

We now have all the ingredients to calculate the quantity we are really interested in: the Kalman gain $\mathbf{K}_k$. In the simplified case, β should minimize the variance of the error of the analyzed variable $\hat{x}$, $\mathrm{E}[\hat{x} - x]^2$. Similarly, $\mathbf{K}_k$ should minimize the variance of the error of the analyzed field, called J:

$$J := \mathrm{E}[(\mathbf{w}_k^a - \mathbf{w}_k^t)^T(\mathbf{w}_k^a - \mathbf{w}_k^t)]. \tag{20}$$

Note the difference with the error covariance matrix of the analyzed field, Eq. (19): the transpose-operator is now at the first factor. We now multiply a row vector by a column vector, which gives a scalar, instead of giving a (covariance) matrix. Note also that J is equal to the trace of $\mathbf{P}_k^a$.

It can be shown that Eq. (19) can be rewritten as:

$$\mathbf{P}_k^a = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_k^b(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)^T + \mathbf{K}_k\mathbf{R}_k\mathbf{K}_k. \quad (21)$$

A proof for this is given in appendix A. Our problem is now transformed to finding $\mathbf{K}_k$ that minimizes $J = \text{tr}(\mathbf{P}_k^a)$. It can also be shown, by using the following matrix calculus identity:

$$\frac{\partial}{\partial A}\,\text{tr}(ABA^T) = 2AB \quad (22)$$

when B is symmetric, that the $\mathbf{K}_k$ that minimizes J is equal to:

$$\mathbf{K}_k^* = \mathbf{P}_k^b\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k^b\mathbf{H}_k^T + \mathbf{R}_k)^{-1}. \quad (23)$$

The proof of this is given in appendix B. This $\mathbf{K}_k^*$ is the Kalman gain, for a given variance in the background field $\mathbf{P}_k^b$ and variance in the observation field $\mathbf{R}_k$.

As a quick summary, the original goal of this section was to combine the background field (model) with the observation field into a new analyzed field. This analyzed field is given by:

$$\mathbf{w}_k^a = \mathbf{w}_k^b + \mathbf{K}_k(\mathbf{w}_k^o - \mathbf{H}_k\mathbf{w}_k^b),$$
$$\text{with } \mathbf{K}_k = \mathbf{K}_k^* = \mathbf{P}_k^b\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k^b\mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

We are now going to apply this method of data assimilation. This is discussed in detail in the following subsections.

8.4 Extra dataset of observations

The previously explained data assimilation scheme requires two types of information: a background field, and an observation field. For the background field, we use the temperature component of the output of the LOVECLIM model, the same as we have been using throughout this whole thesis. For the observation field, we could use the HadCRUT dataset. However, if we combine the model with the HadCRUT dataset, we wouldn't have any reference dataset to compare it to, and we wouldn't have any way to check the quality of the data assimilation. For this reason, we chose to use a different set of measurements as observation field. In fact, we combined two datasets: the ECA dataset with measurements primarily in Europe and Russia, and the USHCN dataset focused purely on the United States of America.

In section 8.4.1 and 8.4.2 we discuss for each extra dataset which measurement stations are used, and how we can prepare the data to be used for data assimilation. This is similar to the preparation of the HadCRUT dataset, discussed in section 3.

8.4.1 ECA

The first extra dataset we use for the data assimilation is the European Climate Assessment dataset (ECA). This project is coordinated by the Dutch KNMI, and features daily measurements of 12 climate variables, like temperature, sea level pressure and precipitation (Klein Tank et al., 2002). These measurements come from a total of 10388 stations in Europe, Russia and the Middle East, although only 3671 are freely available. These stations are shown in Fig. 16a. During this thesis, we only use the temperature variable.

The daily measurements for every station cover different time spans: some stations started their recordings in 1950, some in 1900 and a few all the way back to the 18th century. All stations cover measurements up to at least the year 2000. The respective time spans of the measurements per station are plotted in Fig. 16b.

From the beginning of this thesis, we have only been using the years 1850 to 1998. Fig. 16b shows that the ECA dataset is suitable for this purpose, since many stations have measurements ranging from 1875 to 2000, covering most of the time span of interest.

To be able to use the dataset, we need to transform the daily measurements into monthly averages for every month between 1850 and 1998 (that is, 1788 months in total), and interpolate the station locations to the 64×32 grid with $5.625°$ grid distance. To do this, we need two empty matrices T and N, both $64 \times 32 \times 1788$: that is, for every month, a 64×32 matrix with an element for every grid point. The first matrix, T, contains the accumulated temperatures; the second, N, the number of measurements per grid point and month. Elements of this matrix are called *bins*. Now, we perform the following steps:

1. For every measurement station, we calculate the grid point with smallest distance to the location of the station.
2. For every daily measurement of that station, we

calculate by which month it belongs (for example, measurements from 1 Jan. 1850 to 31 Jan. 1850 go in month 1, or measurements from 1 Feb. 1851 to 28 Feb. 1851 go in month 13, etc.). All invalid measurements (denoted by -9999 in the dataset) are skipped.

3. The value of the first matrix, T, in the bin corresponding to the station grid point and measurement month, is then increased with the measured temperature. Note that we are not yet calculating averages, we are just summing temperatures.
4. The value of the second matrix, N, also in the correct bin, is increased by 1, because there is one more measurement in this specific month / grid point.
5. Once this is done, the monthly averages for every grid point $\overline{T}$, is obtained as $\overline{T}_{ijk} = T_{ijk}/N_{ijk}$, so dividing element wise T by N.

The result is two matrices $\overline{T}$ and N: one containing the monthly averages per grid point of the ECA dataset, the second containing the total number of measurements used to calculate this average, per grid point.

The last step is to create *anomalies*: these are relative temperatures, compared to an average over some given period. Just like for the HadCRUT dataset, we use the average over the years 1961 to 1990.

8.4.2 USHCN

Since the ECA dataset only features measurements over Europe and the Middle East, we combine it with yet another dataset: the United States Historical Climatology Network dataset. This dataset consists of temperature and precipation measurements at 1218 different stations accross the USA (Fig. 16a). The measurements have been processed and checked for duplication, gaps, climatological outliers and temporal and spatial inconsistencies. More information can be found in Menne et al. (2009).

The measurements span a time period from the end of the 19th century up to today. About 95% of the stations have their first measurement in 1893 (Fig. 16b).

Contrary to the ECA dataset, monthly averages were already available and readily calculated, together with the number of measurements used to calculate such monthly averages, per station. We only had to find by which grid point every station belonged, and calculate the anomalies.

8.4.3 Combined extra datasets

Since there is no overlap between the grid points used for the ECA dataset, and those used by the USHCN dataset, we can easily combine the two. We then obtain a new dataset of measurements from 4889 independent stations, spread over Europe, the Middle East and the USA, shown in Fig. 17. This new dataset is used as *observation* field in our data assimilation scheme.

The observations used in Kalman Filter data assimilation are given by an observation field, $\mathbf{w}_k^o$. This is a vector containing all the observations at one time step. However, for every month between 1850 and 1998, the observations are stored in a 64×32 matrix (the longitude is divided in 64 grid points, and the latitude in 32 grid points). This has to be reformatted into a vector with $p = 64 * 32 = 2048$ elements. Now, for every month k, we have an observation field $\mathbf{w}_k^o$ with 2048 elements.

We only need two more ingredient before we can start assimilating the model with the measurements. First, the error covariance matrices $\mathbf{P}_k^b$ of the background field (model) and $\mathbf{R}_k$ of the observation field (measurements). These are discussed in the next section. Second, we need the *observation operator* $\mathbf{H}_k$, which is discussed in section 8.6.

8.5 Error covariance matrices

In the derivation of the Kalman filter, we needed the error covariance matrices $\mathbf{P}_k^b$ and $\mathbf{R}_k$. A covariance matrix $\mathbf{P}$ of a vector $\mathbf{w}$ is a matrix with on it's i, j-th element, the covariance between w_i and w_j.

$$P_{ij} = \text{cov}(w_i, w_j) \tag{24}$$

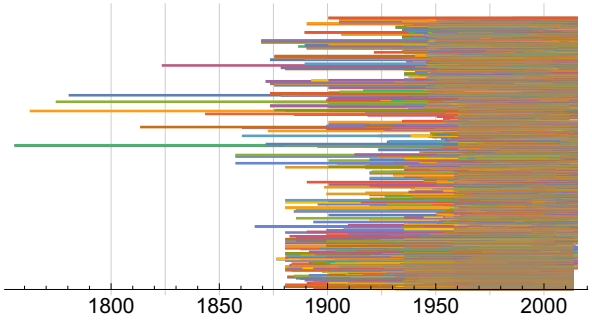
An element P_{ii} on the diagonal of $\mathbf{P}$ contains the *variance* of the element w_i, which we call σ_i^2.

The concrete generation of the error covariance matrix $\mathbf{P}_k^b$ of the background field (the model) is done in a different way than for $\mathbf{R}_k$ for the observation field. We start with the first one, $\mathbf{P}_k^b$.

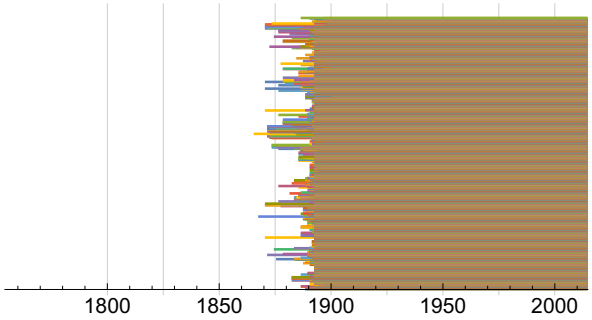
ECA dataset. *The bigger and darker the dot, the more measurements are available.*

USHCN dataset.

(a) Locations of the measurement stations per dataset.



ECA dataset. *The stations are sorted by distance from De Bilt, Netherlands (the top lines are closest to De Bilt).*



USHCN dataset.

(b) Time span of measurements per station. *Every horizontal line corresponds to one of the available measurement stations. The beginning of the line is the date of the first measurement at this station, the end is the date of the last measurement.*

Figure 16: Extra datasets. *On the left, the ECA dataset, on the right, the USHCN dataset.*

8.5.1 Background field covariance matrix

We assume that every element of the background field $\mathbf{w}_k^b$, that is, every grid point in the model output, has a constant variance, σ^2.

The correlation between two grid points is more complicated. We assume that the farther two grid points are apart, the less correlation there is between the two. This can be modelled by exponential decay of the variance, or by more advanced covariance matrices, like described by Gaspari et al. (2006). We have chosen for a simpler model, called AR(3)-model. We start with the simplified case that our background field is a 1-dimensional field, then we extend this to the 2-dimensional grid afterwards.

Consider the background field as in Fig. 18, with i the index of a given grid point. We label the two nearest grid points by 1, the second nearest grid points by 2 and the third nearest grid points by 3.

The AR(3)-model, where AR stands for autoregression, states that the covariance of the value of i with a grid point j is given by:

j	$\text{cov}(i, j) =$
$j = i$	σ^2
$j =$ "1" labeled grid points	$\rho\sigma^2$
$j =$ "2" labeled grid points	$\rho^2\sigma^2$
$j =$ "3" labeled grid points	$\rho^3\sigma^2$
other grid points	0

Figure 17: *Locations of measurement stations of the combined dataset of ECA and USHCN.*

	3	2	1	i	1	2	3	

Figure 18: 1-dimensional grid. *For a given grid point x, the nearest grid points are labeled by 1, the second nearest by 2 and the third nearest by 3.*

where ρ is a constant between 0 and 1. We used $\rho = 0.5$.

The covariance on the grid can thus be represented schematically by Fig. 19.

0	$\rho^3\sigma^2$	$\rho^2\sigma^2$	$\rho\sigma^2$	σ^2	$\rho\sigma^2$	$\rho^2\sigma^2$	$\rho^3\sigma^2$	0

Figure 19: *Covariance on 1-dimensional grid.*

This can be put into a covariance matrix. This is a matrix $\mathbf{P}_k^b$ (Eq. (25)) with on its diagonal the variance of each grid point with itself, σ^2. The first upper and lower diagonal is the correlation with the nearest grid points ($\rho\sigma^2$), the second diagonals with the second nearest grid points ($\rho^2\sigma^2$) and the third diagonals with the third nearest grid points ($\rho^3\sigma$). All the other elements are 0. If there are p elements in the background field, then $\mathbf{P}_k^b$ is a $p \times p$ matrix.

$$\mathbf{P}_k^b = \begin{pmatrix} \sigma^2 & \rho\sigma^2 & \rho^2\sigma^2 & \rho^3\sigma^2 & 0 & \cdots & 0 \\ \rho\sigma^2 & & & & & & \vdots \\ \rho^2\sigma^2 & & & & & & 0 \\ \rho^3\sigma^2 & & & & & & \rho^3\sigma^2 \\ 0 & & & & & & \rho^2\sigma^2 \\ \vdots & & & & & & \rho\sigma^2 \\ 0 & \cdots & 0 & \rho^3\sigma^2 & \rho^2\sigma^2 & \rho\sigma^2 & \sigma^2 \end{pmatrix} \tag{25}$$

This is the final form of the covariance matrix for a 1-dimensional background field. We now extend this to the 2-dimensional grid we are using.

Consider the 2-dimensional background field as in Fig. 20, with (k, l) the index of a given grid point. Just like in the 1-dimensional case, we label the nearest grid points by 1, the second nearest grid points by 2 and the third nearest grid points by 3.

		3a		
	2a	1a	2b	
3d	1d	(k, l)	1b	3b
	2d	1c	2c	
		3c		

Figure 20: *2-dimensional grid. For easier reference, we label the nearest grid points.*

The covariance on the grid now looks like Fig. 21. To put this into a covariance matrix form, we first need to reformat the 2-dimensional 64×32 grid to a 1-dimensional field $\mathbf{w}$. This means that the grid point $(1, 1)$ becomes element 1 of $\mathbf{w}$, grid point $(2, 1)$ becomes element 2 of $\mathbf{w}$, grid point $(64, 1)$ becomes element 64, grid point $(1, 2)$ becomes element 65 and so on, up to grid point $(64, 32)$ which becomes element 2048 of $\mathbf{w}$. This is visualized in Fig. 22. We define the index of the grid point (k, l) in the 1-dimensional field as index i.

Now that the background field is reformatted into a vector $\mathbf{w}_k^b$, we have to find the corresponding covari-

0	0	$\rho^3\sigma^2$	0	0
0	$\rho^2\sigma^2$	$\rho\sigma^2$	$\rho^2\sigma^2$	0
$\rho^3\sigma^2$	$\rho\sigma^2$	σ^2	$\rho\sigma^2$	$\rho^3\sigma^2$
0	$\rho^2\sigma^2$	$\rho\sigma^2$	$\rho^2\sigma^2$	0
0	0	$\rho^3\sigma^2$	0	0

Figure 21: *Covariance on 2-dimensional grid.*

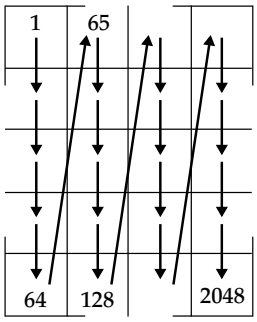


Figure 22: *The globe is divided into 64 parts along the longitude, and 32 parts along the latitude. The* 64×32 *grid is reformatted to a vector, along the longitudinal direction (column-wise).*

ance matrix. For this, we have to take into account the fact that longitudes are periodic. Longitude 0° is the same as longitude 360° (latitudes are not: they go from the South Pole to the North Pole). But first, we consider the grid points outside the boundary layers, that is, any value outside the first two rows and the two last rows shown in figure 22.

We have to find the positions of the "1", "2" and "3" grid points in the reformatted 1-dimensional vector. A careful examination of Fig. 20 and 22 tells us that the indices of the nearest grid points with their respective covariance are given in table 3. This is only true if the index exists (i.e., it is between 1 and 2048).

The second step is to look at the boundary conditions. For example, $i = 128$, like any other grid point, has a non-zero covariance with all grid points around it. However, for $i = 128$, the grid point "below" it (grid point 1c in Fig. 20) is not grid point 129, but instead grid point 65, because of the periodicity of the longitude. We can generalize these boundary conditions for the following set of grid points:

Value of cov. matrix entry:	$\rho\sigma^2$	$\rho^2\sigma^2$	$\rho^3\sigma^2$
Index of cov. matrix entry:	$i-1$ (1a)	$i-65$ (2a)	$i-2$ (3a)
	$i+64$ (1b)	$i+63$ (2b)	$i+128$ (3b)
	$i+1$ (1c)	$i+65$ (2c)	$i+2$ (3c)
	$i-64$ (1d)	$i-63$ (2d)	$i-128$ (3d)

Table 3: *Indices and corresponding values of covariance matrix entries for a 2-dimensional grid, excluding grid points where boundary conditions apply. The position of the grid point in Fig. 20 is shown in brackets.*

- any i such that i mod 64 = 1 (the top row of the 2D grid: 1, 65, 129, etc.)
- any i such that i mod 64 = 2 (the second row of the grid: 2, 66, 130, etc.)
- any i such that i mod 64 = 63 (the second to last raw of the grid: 63, 127, 191, etc.)
- any i such that i mod 64 = 0 (the bottom row of the grid: 64, 128, 192, etc.)

The entries and corresponding values of the covariance matrix for these sets of grid points are listed in table 4. The differences with table 3 are highlighted.

The final error covariance matrix can be constructed row by row. The elements of the i-th row (for i going from 1 to 2048) with their respective covariance are given in table 3 and 4. The first 150 rows and columns of this matrix are shown in Fig. 23, along with a zoom on some more interesting regions, where the effects of the boundary conditions can be seen.

We now have to define the error covariance matrix $\mathbf{R}_k$ of the observation field.

8.5.2 Observation field covariance matrix

Every grid point of the observation field was obtained by averaging multiple measurement stations of the ECA and USHCN datasets. Since the grid has a length scale of approximately 600km (becoming smaller near the poles), we assume that these grid points are so large, that there is no correlation between neighbouring grid points. Neighbouring measurement stations may be correlated, but on the length scale of the grid, we assume this correlation to be negligable.

That means that the error covariance matrix $\mathbf{R}_k$

Value of cov. matrix entry:	$\rho\sigma^2$	$\rho^2\sigma^2$	$\rho^3\sigma^2$
Index of cov. matrix entry:	$i+64$	$i-1$	$i+62$
	$i+64$	$i+127$	$i+128$
	$i+1$	$i+65$	$i+2$
	$i-64$	$i-63$	$i-128$

(a) *First row:* i mod 64 = 1

Value of cov. matrix entry:	$\rho\sigma^2$	$\rho^2\sigma^2$	$\rho^3\sigma^2$
Index of cov. matrix entry:	$i-1$	$i-65$	$i+62$
	$i+64$	$i+63$	$i+128$
	$i+1$	$i+65$	$i+2$
	$i-64$	$i-63$	$i-128$

(b) *Second row:* i mod 64 = 2

Value of cov. matrix entry:	$\rho\sigma^2$	$\rho^2\sigma^2$	$\rho^3\sigma^2$
Index of cov. matrix entry:	$i-1$	$i-65$	$i-2$
	$i+64$	$i+63$	$i+128$
	$i+1$	$i+65$	$i-62$
	$i-64$	$i-63$	$i-128$

(c) *Second to last row:* i mod 64 = 63

Value of cov. matrix entry:	$\rho\sigma^2$	$\rho^2\sigma^2$	$\rho^3\sigma^2$
Index of cov. matrix entry:	$i-1$	$i-65$	$i-2$
	$i+64$	$i+63$	$i+128$
	$i-63$	$i+1$	$i-62$
	$i-64$	$i-127$	$i-128$

(d) *Last row:* i mod 64 = 64

Table 4: *Indices and corresponding values of the covariance matrix for grid points affected by the boundary condition. The values changing from table 3 are highlighted in yellow.*

of the observation field is a simple diagonal matrix, with any off-diagonal elements equal to zero. Like explained before, the diagonal of the matrix contains the variance of every grid point, which depends on the number of measurements per grid station.

This matrix $\mathbf{R}_k$ is different at every time step: there are different number of measurements at every month from Jan. 1850 to Dec. 1998. We first assume that every daily measurement has a constant standard error σ_{dm}. This error could come from human reading errors, interpolation errors and various other sources. A detailed discussion of this can be found in Brohan et al. (2006). We assume that every measurement is a realisation of a random variable with as mean the true temperature, and which is normally distributed with variance σ^2_{dm}. More precisely, we assume that 99% of our measurements lie within 2°C of the true value. By writing this as an equation, we obtain:

$$\int_{-2}^{2} \frac{1}{\sigma_{dm}\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2_{dm}}} \,\mathrm{d}x = 0.99.$$

The solution to this is $\sigma_{dm} = 0.78$, which we now use to calculate the error covariance matrix of the observation field.

For every month k, the standard error of every available grid point i is the standard error of one measurement, divided by the square root of the number of measurements N_k used to calculate the value of that grid point: $\sigma_{i,k} = \sigma_{dm}/\sqrt{N_k}$.

Let M_k be the number of grid points with data available at timestep k. Then for every i from 1 to M_k, the variance per grid point is the standard error squared: $\sigma^2_{i,k} = \sigma^2_{dm}/N_k$.

With this, we can find the final form of $\mathbf{R}_k$:

$$\mathbf{R}_k = \begin{pmatrix} \sigma^2_{1,k} & 0 & \cdots & 0 \\ 0 & \sigma^2_{2,k} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \sigma^2_{M,k} \end{pmatrix}.$$

Note that this matrix is an $M_k \times M_k$ matrix, while the covariance matrix of the background field is an $p \times p$ matrix. Typically, $M_k \ll p$ (which is equivalent to the remark we made in section 8.3.2, that $\dim(\mathbf{w}^o_k) \ll \dim(\mathbf{w}^b_k)$). This is shown in Fig. 24.

We now have almost all the tools needed to perform Kalman filter data assimilation. The only thing needed is the observation operator $\mathbf{H}_k$.

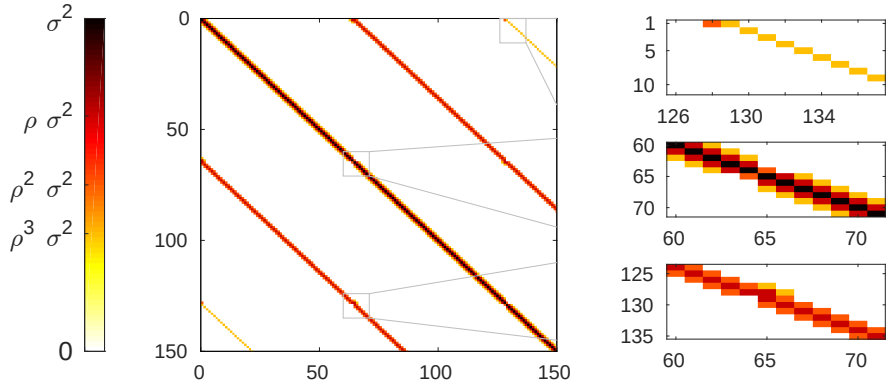


Figure 23: *Covariance matrix $\mathbf{P}_k^b$. For clarity reasons, we only plotted the first 150 rows and columns. Instead of being purely made of constant diagonals, there are irregularities, coming from the boundary conditions. To better view these effects, we added a detailed view of three areas of the matrix.*

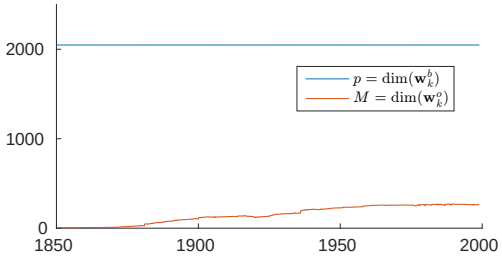


Figure 24: *Dimension of background field compared to the dimension of observation field. Clearly,* $\dim(\mathbf{w}_k^o) \ll \dim(\mathbf{w}_k^b)$.

8.6 Observation operator

Let us, as a reminder, state the final equations used for the data assimilation scheme (Eq. (16) and (23)).

$$\mathbf{w}_k^a = \mathbf{w}_k^b + \mathbf{K}_k(\mathbf{w}_k^o - \mathbf{H}_k\mathbf{w}_k^b),$$
$$\text{with } \mathbf{K}_k = \mathbf{K}_k^* = \mathbf{P}_k^b\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k^b\mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

The observation operator $\mathbf{H}_k$ maps the background field to the observation space.

In our case, both the background field $\mathbf{w}_k^b$ and the observation field $\mathbf{w}_k^o$ are vectors where each element corresponds to a grid point. The only difference is that all p grid points are present in the background field, whereas the observation field only contains a limited set of M_k grid points, corresponding to the grid points where there are existing measurements.

To map the background field to the observation field, we simply remove all the elements corresponding to grid points with no existing measurements in $\mathbf{w}_k^o$. Let $\{i_1^o, i_2^o, ..., i_{M_k}^o\}$ be the set of indices of the grid points of the observations $\mathbf{w}_k^o$. This is the set of indices of the background field we want to keep. In matrix form, the observation operator is an $M_k \times p$ matrix $\mathbf{H}_k$, where the j-th row is made up of zeros, except at the index i_j^o, which is 1, with j ranging from 1 to M_k.

This is best illustrated by an example. Let $\mathbf{w}_k^b$ be a background field with 5 elements: $(a, b, c, d, e)^T$ at grid points $\{1, 2, 3, 4, 5\}$, and let $\mathbf{w}_k^o$ be an observation field with 3 elements: $(\kappa, \lambda, \mu)^T$ at grid points $\{2, 4, 5\}$. Then $\mathbf{H}_k$ is the 3×5 matrix given by:

$$\mathbf{H}_k = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

This gives us the background field mapped to the observation space:

$$\mathbf{w}_k^o - \mathbf{H}_k\mathbf{w}_k^b = \begin{pmatrix} \kappa \\ \lambda \\ \mu \end{pmatrix} - \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \\ e \end{pmatrix}$$
$$= \begin{pmatrix} \kappa - b \\ \lambda - d \\ \mu - e \end{pmatrix}.$$

It should be noted that the observation operator

formally maps the background field to the raw observations, in our case, the daily measurement from the stations. Therefore, the observation operator actually also encompasses the monthly averaging and the interpolation to nearest grid points. We call the averaging operator $\mathbf{A}_k$, and the interpolation to grid operator $\mathbf{G}_k$. The total observation operator now becomes:

$$\mathbf{H}_{k,\text{eff}} = \mathbf{A}_k\mathbf{G}_k\mathbf{H}_k := \widetilde{\mathbf{G}}_k\mathbf{H}_k,$$

where $\widetilde{\mathbf{G}}_k$ is the operator combining averaging and interpolation.

Let $\widetilde{\mathbf{w}}_k^o$ be the field of raw observations, where each element corresponds to the observation of a different measurement station at time k. When we consider these raw observations, the Kalman filter equations also change. The observation operator is now $\widetilde{\mathbf{G}}_k\mathbf{H}_k$, and the observation error covariance matrix now gives the covariance per measurement station. We call this matrix $\widetilde{\mathbf{R}}_k$. The Kalman filter becomes:

$$\mathbf{w}_k^a = \mathbf{w}_k^b + \widetilde{\mathbf{K}}_k(\widetilde{\mathbf{w}}_k^o - \widetilde{\mathbf{G}}_k\mathbf{H}_k\mathbf{w}_k^b), \text{ with} \tag{26}$$

$$\widetilde{\mathbf{K}}_k = \mathbf{P}_k^b(\widetilde{\mathbf{G}}_k\mathbf{H}_k)^T\left((\widetilde{\mathbf{G}}_k\mathbf{H}_k)\mathbf{P}_k^b(\widetilde{\mathbf{G}}_k\mathbf{H}_k)^T + \widetilde{\mathbf{R}}_k\right)^{-1}. \tag{27}$$

Let us now assume that $\widetilde{\mathbf{G}}_k$ is orthonormal ($\widetilde{\mathbf{G}}_k\widetilde{\mathbf{G}}_k^T = I$). We will come back to this assumption later on. With this, we can rewrite Eq. (26) to:

$$\begin{aligned}\mathbf{w}_k^a &= \mathbf{w}_k^b + \widetilde{\mathbf{K}}_k(\widetilde{\mathbf{G}}_k\widetilde{\mathbf{G}}_k^T\widetilde{\mathbf{w}}_k^o - \widetilde{\mathbf{G}}_k\mathbf{H}_k\mathbf{w}_k^b)\\ &= \mathbf{w}_k^b + \widetilde{\mathbf{K}}_k\widetilde{\mathbf{G}}_k(\widetilde{\mathbf{G}}_k^T\widetilde{\mathbf{w}}_k^o - \mathbf{H}_k\mathbf{w}_k^b).\end{aligned} \tag{28}$$

The next step is to calculate $\widetilde{\mathbf{K}}_k\widetilde{\mathbf{G}}_k$:

$$\begin{aligned}\widetilde{\mathbf{K}}_k\widetilde{\mathbf{G}}_k &= \mathbf{P}_k^b(\widetilde{\mathbf{G}}_k\mathbf{H}_k)^T\left((\widetilde{\mathbf{G}}_k\mathbf{H}_k)\mathbf{P}_k^b(\widetilde{\mathbf{G}}_k\mathbf{H}_k)^T + \widetilde{\mathbf{R}}_k\right)^{-1}\widetilde{\mathbf{G}}_k\\ &= \mathbf{P}_k^b\mathbf{H}_k^T\widetilde{\mathbf{G}}_k^T\left((\widetilde{\mathbf{G}}_k\mathbf{H}_k)\mathbf{P}_k^b\mathbf{H}_k^T\widetilde{\mathbf{G}}_k^T + \widetilde{\mathbf{R}}_k\right)^{-1}\widetilde{\mathbf{G}}_k\\ &= \mathbf{P}_k^b\mathbf{H}_k^T(\widetilde{\mathbf{G}}_k)^{-1}\left((\widetilde{\mathbf{G}}_k\mathbf{H}_k)\mathbf{P}_k^b\mathbf{H}_k^T\widetilde{\mathbf{G}}_k^T + \widetilde{\mathbf{R}}_k\right)^{-1}(\widetilde{\mathbf{G}}_k^T)^{-1}.\end{aligned}$$

In the last step, we use the fact that $\widetilde{\mathbf{G}}_k^T = \widetilde{\mathbf{G}}_k^{-1}$, and that $\widetilde{\mathbf{G}}_k = (\widetilde{\mathbf{G}}_k^{-1})^{-1} = (\widetilde{\mathbf{G}}_k^T)^{-1}$. Now, since we assumed that $\widetilde{\mathbf{G}}_k$ was orthonormal, it is also square and invertible. This means we can use the identity $A^{-1}B^{-1} = (BA)^{-1}$ for A and B square and invertible matrices. This allows us to further rewrite $\widetilde{\mathbf{K}}_k\widetilde{\mathbf{G}}_k$ to:

$$\begin{aligned}\widetilde{\mathbf{K}}_k\widetilde{\mathbf{G}}_k &= \mathbf{P}_k^b\mathbf{H}_k^T(\widetilde{\mathbf{G}}_k^T\widetilde{\mathbf{G}}_k\mathbf{H}_k\mathbf{P}_k^b\mathbf{H}_k^T\widetilde{\mathbf{G}}_k^T\widetilde{\mathbf{G}}_k + \widetilde{\mathbf{G}}_k^T\widetilde{\mathbf{R}}_k\widetilde{\mathbf{G}}_k)^{-1}\\ &= \mathbf{P}_k^b\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k^b\mathbf{H}_k^T + \widetilde{\mathbf{G}}_k^T\widetilde{\mathbf{R}}_k\widetilde{\mathbf{G}}_k)^{-1}.\end{aligned}$$

By renaming $\widetilde{\mathbf{K}}_k\widetilde{\mathbf{G}}_k$ to $\mathbf{K}_k$ and $\widetilde{\mathbf{G}}_k^T\widetilde{\mathbf{R}}_k\widetilde{\mathbf{G}}_k$ to $\mathbf{R}_k$, we obtain the original formula for the Kalman gain, without considering the interpolation. The matrix $\widetilde{\mathbf{G}}_k^T\widetilde{\mathbf{R}}_k\widetilde{\mathbf{G}}_k$ is then the error covariance matrix of the raw observations averaged and interpolated to the grid points. However, for this to work, we have to assume that $\widetilde{\mathbf{G}}_k$ is orthonormal, and therefore square. This is not the case, since there are already many more measurement stations (4889) than there are grid points (2048). This means that we should use Eq. (26) and (27) instead of Eq. (16) and (23). However, we neglect the errors induced by averaging and interpolation, and use Eq. (16) and (23).

It is now finally time to perform the actual Kalman filter data assimilation. The results are discussed in the next section.

8.7 Results of offline data assimilation

In the theoretical description of the data assimilation scheme we are using, we showed that the assimilation process comes back to two equations (Eq. (16) and (23)):

$$\begin{aligned}\mathbf{w}_k^a &= \mathbf{w}_k^b + \mathbf{K}_k(\mathbf{w}_k^o - \mathbf{H}_k\mathbf{w}_k^b),\\ \text{with } \mathbf{K}_k &= \mathbf{K}_k^* = \mathbf{P}_k^b\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k^b\mathbf{H}_k^T + \mathbf{R}_k)^{-1}\end{aligned}$$

In section 8.4.1 and 8.4.2, we discussed how we obtain the observation field $\mathbf{w}_k^o$, by combining the ECA and USHCN datasets. In section 8.5, we discussed in detail how to calculate the error covariance matrices $\mathbf{P}_k^b$ (which still depends on the parameter σ) and $\mathbf{R}_k$. Finally, in section 8.6, we showed a method to obtain the observation operator $\mathbf{H}_k$. With these sections combined, we are ready to perform the data assimilation.

As mentioned before, this data assimilation considers every time moment k separately. Observations at time k only influence the temperatures of the background field of time k. In Fig. 25, we show the resulting assimilated field $\mathbf{w}_k^a$ for one of these moments (the year 1975) with $\sigma = 0.1$, along with the background and observation fields $\mathbf{w}_k^b$ and $\mathbf{w}_k^o$. For this year, the biggest differences between $\mathbf{w}_k^b$ and $\mathbf{w}_k^a$ are around Scandinavia and the centre of the USA. We see in Fig. 25 that roughly speaking, the analyzed field is equal to the observations where the observations exist, and to the background field where there are no observations available.

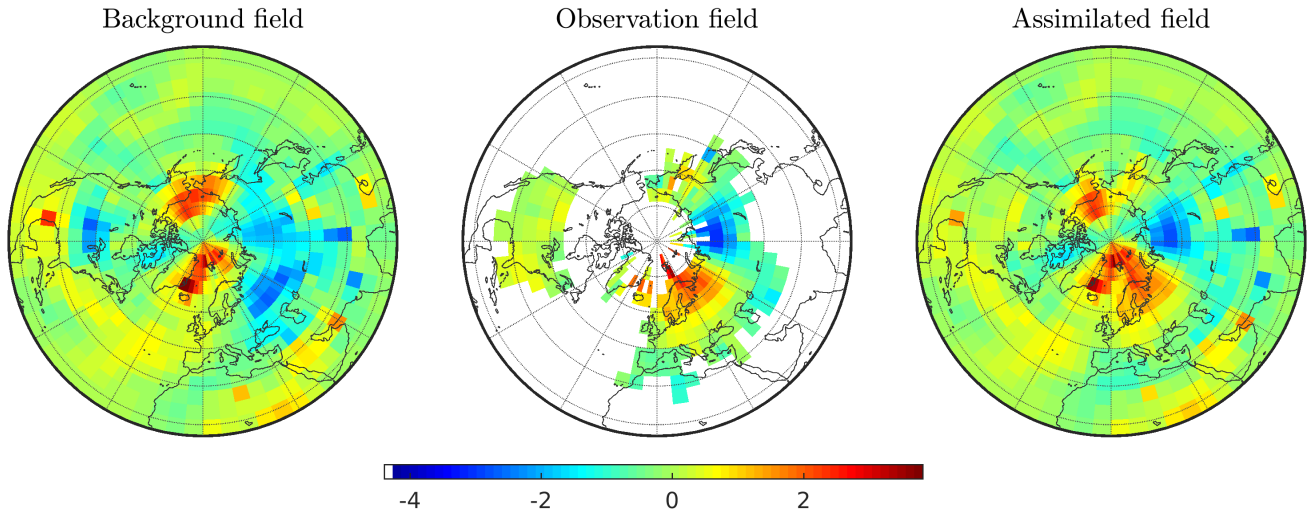


Figure 25: *Example of offline data assimilation, using the background and observation anomaly fields of 1975 and with $\sigma = 0.1$. The data show warmer temperatures around Scandinavia and the centre of the USA than the model has reproduced.*

Now that we have new assimilated anomaly fields, we can compare them to the original HadCRUT dataset, and check if they agree better than without data assimilation. This is done in the next section.

8.8 Error estimation using data assimilation

Data assimilation allows us to create an anomaly field which better represents the true anomalies between 1850 and 1998. This is done by combining the observation field, derived from the ECA and USHCN datasets, with the background field, which is the temperature representation of the LOVECLIM model.

In the data assimilation method described in the previous section, we didn't explicitly give the value of the standard deviation of the model anomalies, σ. It is a measure of the uncertainty of the model. Since we don't know the value of σ, we perform the data assimilation process for different values: $\sigma = 0.01$, 0.05, 0.1, 0.25, 0.5, 1 and 2. The method we use to choose which σ corresponds best to the model is, just like we have done in section 7, by using the PC projection method.

For this method, we use the assimilated matrix X_{assim} and the data matrix X_{data}. Note that the k-th row of X_{assim} is equal to $\mathbf{w}_k^a$. In section 7, we used X_{model} instead of X_{assim}. We can split X_{assim} and X_{data} into their respective PC and EOF components:

$$\begin{cases} X_{\text{assim}} = C_{\text{assim}} U_{\text{assim}}^T \\ X_{\text{data}} = C_{\text{data}} U_{\text{data}}^T. \end{cases}$$

Just like in section 7, the assimilated field is projected onto the EOFs of the HadCRUT data (previously *model on data*, now *assim. on data*). Since we have seen that *model on data* gives higher correlations than *data on model*, we do not consider the *data on assim.* case. Mathematically, this gives us new descriptions of X_{assim} and X_{data}:

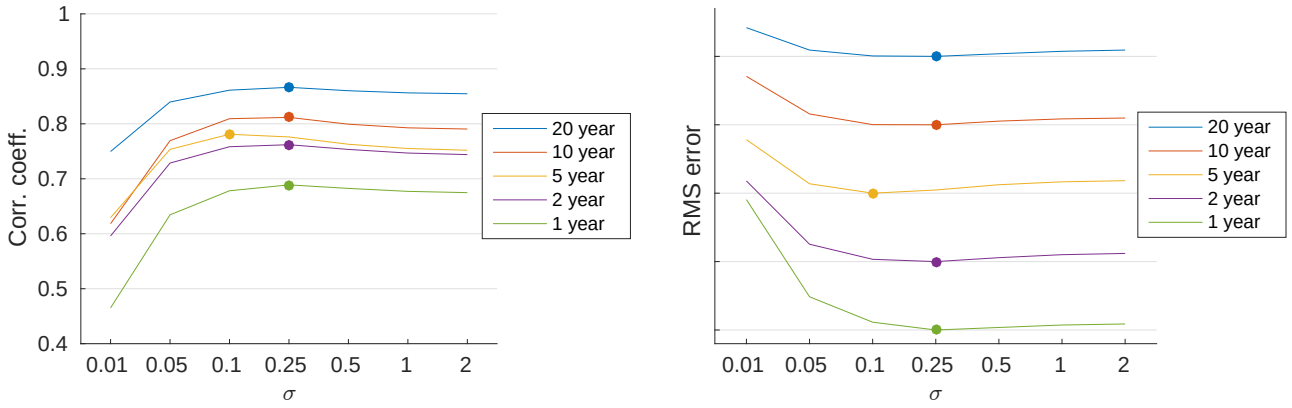
$$\begin{cases} X_{\text{assim}} = \widehat{C}_{\text{assim}} U_{\text{data}}^T \\ X_{\text{data}} = C_{\text{data}} U_{\text{data}}^T \end{cases}$$

We can now compare the original PC of one matrix with the projected PC of the other:

$$\begin{cases} \widehat{C}_{\text{assim}} = X_{\text{assim}} U_{\text{data}} \\ C_{\text{data}} \end{cases}. \tag{29}$$

We use two measures of fit for the comparison of these PCs: first the correlation coefficient $\rho(C1, C2)$, and second the RMS error between the PCs.

To obtain these results, we first average the anomalies over a period of 1, 2, 5, 10 and 20 years, and perform data assimilation with different values of the model standard deviation: $\sigma = 0.01$, 0.05, 0.1, 0.25, 0.5, 1 and 2. Then the PCs are calculated as



(a) *Correlation coefficients*

(b) *RMS errors. Since only the relative values between various values of σ are important, these are in arbitrary units.*

Figure 26: Correlation coefficients and RMS errors. *First, the anomalies are averaged over 1, 2, 5, 10 and 20 years. Then we perform data assimilation for multiple value of σ. On the left, the correlation between between the projected first Principal Component $\widehat{C}_{assim}$ and C_{data} is calculated, on the right their RMS errors. The values of σ with maximum correlation or lowest RMS error are highlighted.*

in Eq. (29). Some of these PCs are shown in Fig. 27. Next, the correlation between them are calculated, visible in Fig. 26a. From these coefficients, we see that the assimilated fields which have highest correlations with the HadCRUT PC, are the ones corresponding to $\sigma = 0.1$ for the 5 year average and to $\sigma = 0.25$ for the 1, 2, 10 and 20 year averages. Since for this thesis, we consider the HadCRUT dataset as the *true* anomalies, it would seem that the standard deviation of the temperature of the model is between $\sigma = 0.1$ and $\sigma = 0.25$.

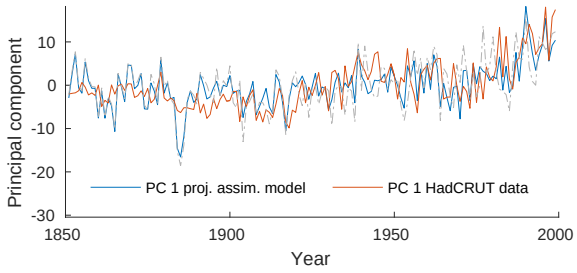
The next measure of skill is the RMS error between the two PCs $\widehat{C}_{\text{assim}}$ and C_{data}, with $RMS(\mathbf{x}) := \sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}$ the root mean square value of a vector $\mathbf{x}$. This error also tells us for which σ the assimilated field PC is closest to the HadCRUT PC. For this, we only have to know the value of σ for which the error is minimum. Therefore, we show these values without unit nor scale, in Fig. 26b. This measure of skill gives the same results: the optimal σ is between 0.1 and 0.25.

It should be noted that these results strongly depend on the assumptions previously made (shape of $\mathbf{P}_k^b$, value of σ_{dm}, accuracy of HadCRUT dataset...). However, this method shows that it is possible to estimate the variance σ^2 of the temperature of the model using data assimilation and EOF decomposition.

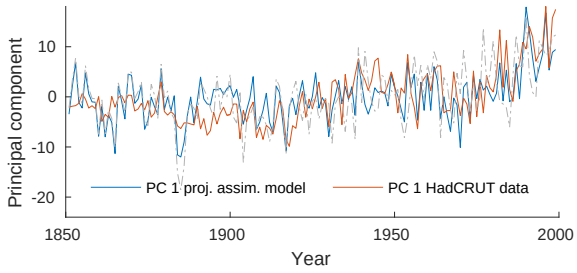
When using data assimilation in this section 8, we have only considered one month (or year, or 10 years...) at a time. Observations at moment k did not influence observations at moment k' ($k \neq k'$). In the next section, we develop a methodology to perform data assimilation, while using information about every moment.

9 Data assimilation on EOFs

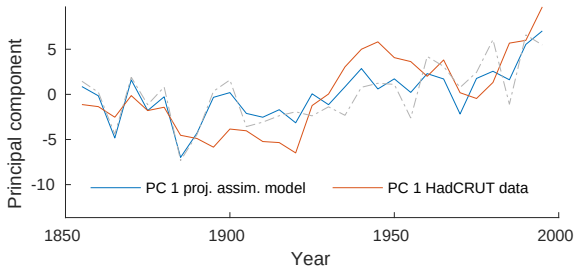
We have shown in section 8 that Kalman filter data assimilation is a suitable way to combine model output with observations, without having to run the model again. The downside, however, is that the assimilation process at moment k only considers observations at moment k. It would be interesting to be able to use all the information available, instead of just time k. In section 5 and 6, we have discussed how to decompose a dataset into patterns of highest variability, using EOF and DINEOF decomposition. In this section, we aim to combine the two principles: we perform data assimilation directly on EOFs.
For this, we use the same method as in section 8: the



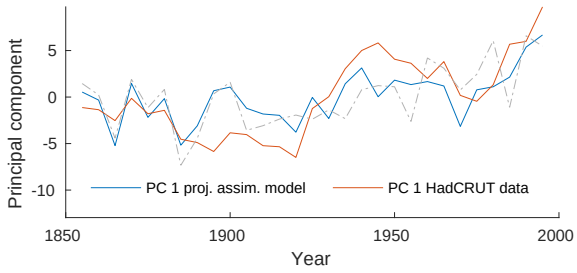
(a) *Yearly*, $\sigma = 0.1$



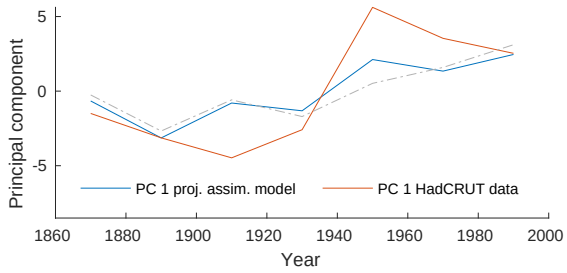
(b) *Yearly*, $\sigma = 1$



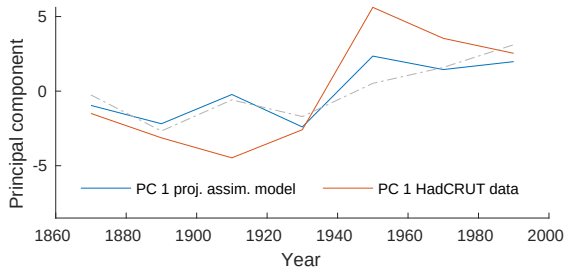
(c) *5 years*, $\sigma = 0.1$



(d) *5 years*, $\sigma = 1$



(e) *20 years*, $\sigma = 0.1$



(f) *20 years*, $\sigma = 1$

Figure 27: First principal component *of the projected assimilated field and the HadCRUT dataset. Fig. (a) and (b) use yearly averages, (c) and (d) use 5 year average and (e) and (f) use 20 year average. The left column was calculated using* $\sigma = 0.1$*, the right using* $\sigma = 1$*. The dashed gray line is the reference, unassimilated model output projected onto the HadCRUT EOFs.*

Kalman filter, given by Eq. (16) and (23):

$$\mathbf{w}_k^a = \mathbf{w}_k^b + \mathbf{K}_k(\mathbf{w}_k^o - \mathbf{H}_k\mathbf{w}_k^b),$$
$$\text{with } \mathbf{K}_k = \mathbf{K}_k^* = \mathbf{P}_k^b\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k^b\mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

Since we are assimilating two EOFs instead of regular observation and background fields, this will need some adjustments. These will be discussed in the following steps:

- We limit the spatial domain to the part of the Earth where we have most measurements (section 9.1);
- The model output and data are decomposed using EOF/DINEOF, and the data is projected onto the PC of the model anomalies (section 9.2);
- We calculate error covariance matrices for the model and the observations (section 9.3);
- Data assimilation is performed using the Kalman filter (section 9.4);
- The analyzed field is reconstructed by projecting the assimilated EOF onto the original PC (section 9.5);
- Finally, the quality of this analyzed field is investigated using the PC projection analysis first discussed in section 7 (section 9.6).

When describing this methodology, we make many assumptions. Although we try to give them a physical or mathematical underpinning as much as possible, many of these assumptions would require more thorough research. The goal of this section, however, is to show the possibility to perform data assimilation on EOFs.

9.1 Data used

Just like in the "regular" data assimilation of section 8, we combine the model anomalies with the data of the ECA and USHCN datasets (see sections 8.4.1 to 8.4.3). Their measurement stations are centered around the USA, Europe and Russia. We therefore restrict our spatial domain to all the grid points with latitudes between $30°N$ and $70°N$. Note that if we chose the whole Northern Hemisphere, it would be more difficult to calculate the EOFs of the observation field, since their would be many more missing values. This was not necessary in section 8, where we simply restricted the domain to the Northern Hemisphere.

The spatial domain we use for this section is plotted in Fig. 28. With this new domain, we have 64×8 grid

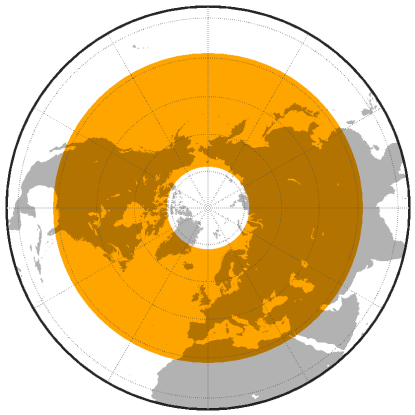


Figure 28: *Spatial domain used for data assimilation on EOFs, highlighted in orange. This covers all grid points with latitude between $30°N$ and $70°N$, and all longitudes.*

points for the model anomalies and for the observations. Before we can perform the EOF decomposition, we first have to multiply every anomaly by the square root of the cosine of its latitude, to account for the surplus of grid points at higher latitudes (see section 5.5).

The last preparation step is to average the anomalies over time periods of 1 year, 2, 5, 10 and 20 years.

We now have two datasets ready to be used for EOF decomposition and data assimilation: X_{model} and X_{data}. We also apply this domain limitation, weighting and averaging to the HadCRUT dataset: we will use this data as reference in section 9.6. The next step is to decompose these datasets into EOF/PCs.

9.2 EOF / DINEOF decomposition

The anomaly fields X_{model} and X_{data} can be decomposed into spatially dependent parts, the EOFs, and time-dependent parts, the PCs. In section 5.4, we have seen that these can be calculated using Singular Value Decomposition:

$$X = A\ \Gamma\ U^T$$
$$:= C\ U^T,$$

where we defined the *principal component* as $C := A\ \Gamma$, and the EOF as U^T. In section 7, we kept the spatial component fixed. We projected one set of anomalies onto the EOF of the other dataset, this way obtaining two time-dependent PCs C_{model} and $\widehat{C}_{\text{data}}$ projected

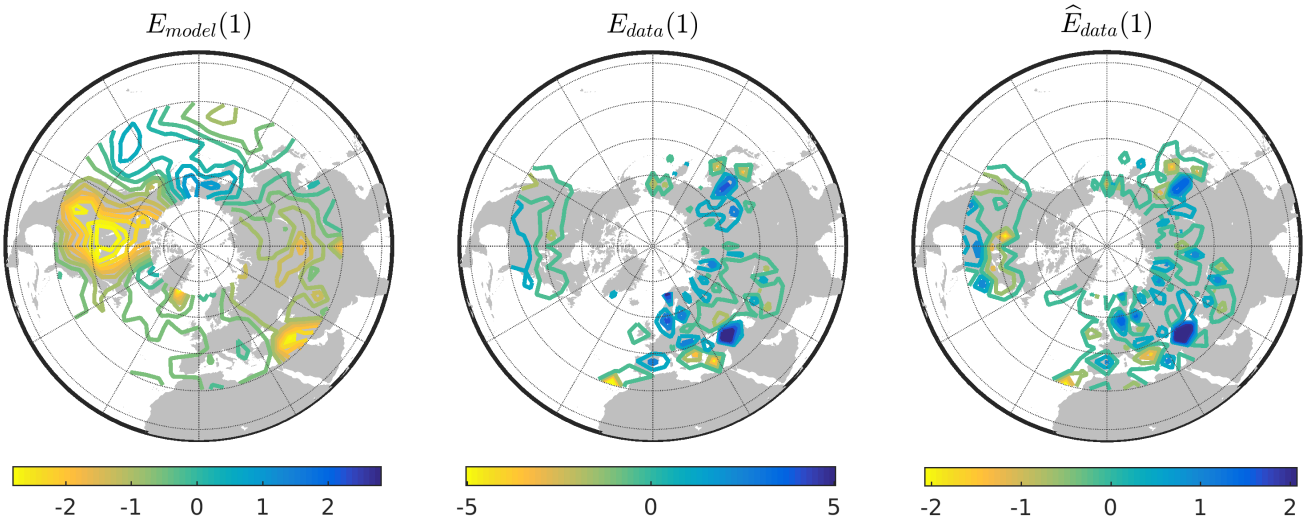


Figure 29: *EOFs computed over the limited domain, using the new definition of the EOF. The left figure shows the first EOF of the model field. The middle figure is the first EOF obtained using the DINEOF decomposition, which also gives a reconstructed field. This field is projected onto the PC of the model anomalies, and is shown in the right figure.*

onto the same EOF:

$$\begin{cases} X_{\text{model}} = C_{\text{model}}\, U^T_{\text{model}} \\ X_{\text{data}} = \widehat{C}_{\text{data}}\, U^T_{\text{model}} \end{cases},$$

with $\widehat{C}_{\text{data}} := X_{\text{data}} U_{\text{model}}$.

We now want to do the opposite: since we are assimilating the space-dependent EOF, we have to keep the time dependency fixed. Instead of projecting onto an EOF, we now project onto a PC. A naïve, and wrong, implementation of this would be by simply multiplying X_{data} by the PC of the model anomalies, C_{model}, instead of its EOF U_{model}:

$$\begin{cases} X_{\text{model}} = C_{\text{model}}\, U^T_{\text{model}} \\ X_{\text{data}} = C_{\text{model}}\, \widehat{U}^T_{\text{data}} \end{cases}, \tag{30}$$

with $\widehat{U}_{\text{data}} := X^T_{\text{data}} C_{\text{model}}$. The reason this is wrong becomes clear when filling $\widehat{U}_{\text{data}}$ into Eq. (30), and using the definition of C_{model}:

$$\begin{aligned} C_{\text{model}}\, \widehat{U}^T_{\text{data}} &= C_{\text{model}}\, (X^T_{\text{data}} C_{\text{model}})^T \\ &= C_{\text{model}}\, C^T_{\text{model}}\, X_{\text{data}} \\ &= (A_{\text{model}}\, \Gamma_{\text{model}})\, (A_{\text{model}}\, \Gamma_{\text{model}})^T\, X_{\text{data}} \\ &= A_{\text{model}}\, (\Gamma_{\text{model}}\, \Gamma^T_{\text{model}})\, A^T_{\text{model}}\, X_{\text{data}}. \end{aligned}$$

While it is true that $(A_{\text{model}} A^T_{\text{model}}) = I$, since A is orthonormal, this is not true for Γ. Γ is a diagonal matrix with the singular values of X on its diagonal. The last equation cannot be further simplified. Hence, $X_{\text{data}} \neq C_{\text{model}} \widehat{U}^T_{\text{data}}$. We need another way to keep the time-dependent component fixed.

9.2.1 New definition of PC/EOFs

In order to still be able to keep the time-dependent component fixed, we have to redefine the principal component and EOFs. From SVD, we can write X as $X = A\, \Gamma\, U^T$. Since Γ is diagonal, it simply multiplies the columns of A or the rows of U^T by a constant. Instead of defining the principal component as $C := A\, \Gamma$ and the EOF as U^T, we now define a new principal component and EOF:

$$\begin{aligned} \text{PC}: &\quad A, \\ \text{EOF}: &\quad E^T := \Gamma\, U^T. \end{aligned}$$

With this definition, X can be written as $X = A\, E^T$. Since the new principal component, E, is now orthonormal, we can project X_{data} onto the EOFs of X_{model}:

$$\begin{cases} X_{\text{model}} = A_{\text{model}}\, E^T_{\text{model}} \\ X_{\text{data}} = A_{\text{model}}\, \widehat{E}^T_{\text{data}} \end{cases}, \tag{31}$$

with $\widehat{E}_{\text{data}} := X_{\text{data}}^T A_{\text{model}}$. In fact, since A_{model} is orthonormal by definition, we find:

$$\begin{aligned} A_{\text{model}}\, \widehat{E}_{\text{data}}^T &= A_{\text{model}}\, (X_{\text{data}}^T A_{\text{model}})^T \\ &= (A_{\text{model}}\, A_{\text{model}}^T)\, X_{\text{data}} \\ &= X_{\text{data}}. \end{aligned}$$

By applying this with the model anomalies and observations from ECA/USHCN, we can calculate the EOFs. When calculating the projected EOF $\widehat{E}_{\text{data}} = X_{\text{data}}^T A_{\text{model}}$, we first reconstruct the observation field X_{data} using the DINEOF reconstruction. The results of this are plotted in Fig. 29.

9.3 Error covariance matrices

Just like in section 8, we need to define an error covariance matrix for the "background" and "observation" field. Now, these matrices describe the spatial covariance of an EOF. We split the discussion in two parts: first, the background EOF covariance matrix is defined in section 9.3.1, than the observation EOF covariance matrix in section 9.3.2.

9.3.1 Background EOF covariance matrix

An EOF is a spatial pattern, in our case, of anomalies. We assume that the covariances behave like a normal anomaly field: the closest grid points have highest covariance, grid points farther away have less. For this reason, we use the same AR(3)-model as defined in section 8.5.1. If a grid point of the EOF with index i has variance σ_i^2, then the nearest grid points have covariance $\rho\sigma_i^2$, the second nearest grid points $\rho^2\sigma_i^2$, the third nearest $\rho^3\sigma_i^2$, and any other grid point zero covariance with i. We now need an estimate for the variance of the anomalies per grid point. We define the *uncertainty*, also called standard deviation, as the square root of the variance.

In section 8, we assumed this variance to be constant for any grid point. This variance comes from the model itself, and is still present when considering EOFs. We call this $\sigma_{i,\text{model}}^2$. There is, however, another source of variance: the EOF decomposition also introduces an uncertainty for every grid point: we call this variance $\sigma_{i,\text{EOF}}^2$.

There are various ways to estimate $\sigma_{i,\text{EOF}}$. Björnsson (1997) describes a method using Monte Carlo simulations by first randomizing the time component of the anomaly field, then perform EOF decomposition on it. By repeating this many times, spatial correlations can be calculated. We use another method, often referred to as the rule of thumb described by North et al. (1982). This method gives an estimate of the uncertainty in the eigenvalues and eigenvectors of the covariance matrix of the anomalies (i.e., the EOFs). Let λ_m^2 be the m-th eigenvalue, corresponding to the m-th EOF $\mathbf{u}_m$. Then, the rule of thumb states that the uncertainty in the eigenvalue, $\Delta\lambda_m^2$, is given by:

$$\Delta\lambda_m^2 \approx \lambda_m^2 \sqrt{\frac{2}{n^*}}, \tag{32}$$

where n^* is the number of independent measurements or observations used to calculate the EOF at grid point i. Here, we assume n^* for the model to be simply equal to n, which is the number of timesteps used (for example, $n = 1788$ for monthly averages and $n = 15$ for 10 year averages). The uncertainty in the EOF, $\Delta\mathbf{u}_m$, is then given by:

$$\Delta\mathbf{u}_m \approx \frac{\Delta\lambda_m^2}{\lambda_m^2 - \lambda_q^2}\mathbf{u}_q. \tag{33}$$

In this definition, λ_q^2 is the *closest* eigenvalue to λ_m^2, in absolute difference, and $\mathbf{u}_q$ its corresponding EOF. Since we only consider the first EOF, the closest eigenvalue is always the second eigenvalue.

The method described in North et al. (1982) states that the uncertainty in the first EOF is proportional to the values of the second EOF. Note that we used $\mathbf{u}$ for the EOF: this refers to the original definition of an EOF, described in section 5. In this section, we defined the EOFs to be equal to the rows of ΓU^T. In other words, the original m-th EOF is multiplied by γ_m, the m-th value of the diagonal of Γ. As we have shown in section 5.4, this is equal to:

$$\gamma_m = \lambda_m \sqrt{n}.$$

The uncertainty of the new EOF is therefore equal to the uncertainty of the original EOF, multiplied by γ_m:

$$\sigma_{i,\text{EOF}} := \Delta\mathbf{e}_m = \lambda_m \sqrt{n} \Delta\mathbf{u}_m.$$

We assume that the same argument holds for the uncertainty coming from the model. If the model has a standard deviation σ, then the value of $\sigma_{i,\text{model}}$ becomes:

$$\sigma_{i,\text{model}} := \lambda_m \sqrt{n}\sigma.$$

In section 8.8, we found that the best estimate for the standard deviation of the model is approximately $\sigma = 0.2$. We will use this value during the rest of this section.

The total variance of the i-th grid point is equal to $\sigma_i^2 = \sigma_{i,\text{model}}^2 + \sigma_{i,\text{EOF}}^2$. For simplicity, we assume that the model error and the error of EOF decomposition are independent and can therefore be added together. The values of $\sigma_{i,\text{model}}$ and $\sigma_{i,\text{EOF}}$ are shown in Fig. 30a and 30c. The error covariance matrix can now be constructed in the same way as described in section 8.5.1. The difference here is that the diagonal is not constant. This kind of covariance matrix is called *heteroskedastic*. Care has to be taken to construct a matrix that is still symmetric. Instead of constructing the matrix row by row, we construct the i-th row and directly use the transpose of the newly constructed row as the i-th column of the matrix. We call the newly constructed error covariance matrix of the model EOF $\mathbf{P'}^b_m$, where the prime notation is used to distinguish it from the error covariance matrix defined in section 9.3.1.

9.3.2 Observation EOF covariance matrix

A similar method as described in the previous section can be applied to create the error covariance matrix of the observation EOF. Since we consider the EOFs of the observations, we cannot assume that the grid points are uncorrelated, like we have done in section 8.5.2. Here, we also use the AR(3)-model for the error covariance matrix.

Like in the previous section, the variance of the observation EOF is made of a component coming from the uncertainties in the measurements, and a component coming from the uncertainties of the EOF. Let $\widetilde{\sigma}_i^2$ be the total variance of the observation EOF at grid point i, and let $\widetilde{\sigma}_{i,\text{meas}}^2$ and $\widetilde{\sigma}_{i,\text{EOF}}^2$ be the variance coming respectively from the measurements and the EOF decomposition.

We first discuss $\widetilde{\sigma}_{i,\text{meas}}$. This uncertainty is equal to the uncertainty of a single measurement, $\sigma_{dm} = 0.78$, divided by the square root of N_i. In section 8.5.2, $N_i = N_{k,i}$ was simply the number of measurements at a certain grid point i during the k-th period (either the k-th month, the k-th year, 10 years, etc.). Here, we are still averaging over these periods. It is however not possible to know which period we should use to count the number of measurements. It is also not physical to simply count the number of measurements of the whole period from 1850 to 1998: this is independent of the averaging period. For this reason, we define N_i to be the average of all values of $N_{k,i}$:

$$N_i := \frac{1}{n}\sum_{k=1}^{n} N_{k,i},$$

where we sum over the n periods (for 1 yearly averages, there are $n = 149$ periods). Using this, we find an expression for $\widetilde{\sigma}_{i,\text{meas}}$:

$$\widetilde{\sigma}_{i,\text{meas}} := \frac{\sigma_{dm}}{\sqrt{N_i}}.$$

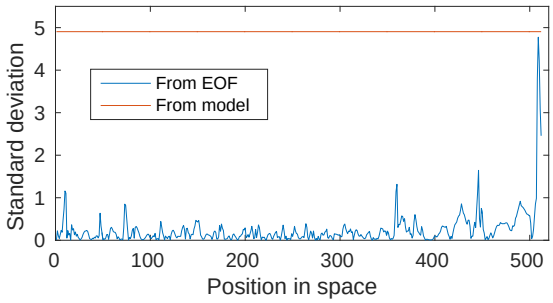
There are, however, many grid points without any observation during the whole period from 1850 to 1998. We assume that the variance at these points is infinite. Since computationally this is not possible, we restrict the observation space to the set of grid points with at least one observation. We will come back to this in section 9.4.

The second component of the uncertainty in the observation EOF comes from the EOF decomposition itself. We also use the rule of thumb proposed by North et al. (1982):

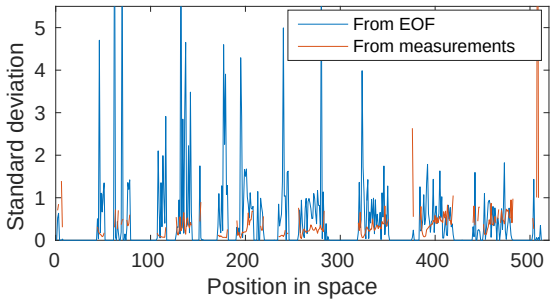
$$\Delta\widetilde{\lambda}_m^2 \approx \widetilde{\lambda}_m^2\sqrt{\frac{2}{\widetilde{n}^*}},$$
$$\Delta\widetilde{\mathbf{u}}_m \approx \frac{\Delta\widetilde{\lambda}_m^2}{\widetilde{\lambda}_m^2 - \widetilde{\lambda}_q^2}\widetilde{\mathbf{u}}_q.$$

Here, $\widetilde{\lambda}^2$ and $\widetilde{\mathbf{u}}$ are the eigenvalues and EOFs of the observation field, and $\widetilde{n}^*$ is the *effective sample size*, usually a certain fraction of the number of time steps, n. For the model, we used $n^* = n$. Here, we use the number of timesteps with existing measurements, averaged over every grid point. A more elaborate method is discussed in Thiébaux & Zwiers (1984). This yields the values of $\widetilde{n}^*$ shown in table 5. These values are rounded to the nearest integer, since we only consider full years.

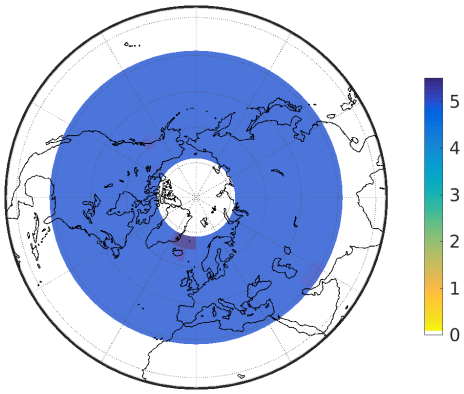
This rule of thumb doesn't explicitly take into account the fact that we use DINEOF reconstruction for the observations, because of the missing data. However, grid points without any measurements have infinite variance in $\widetilde{\sigma}_{i,\text{meas}}$. Grid points with some missing values also have a lower $\widetilde{n}^*$. This way, while



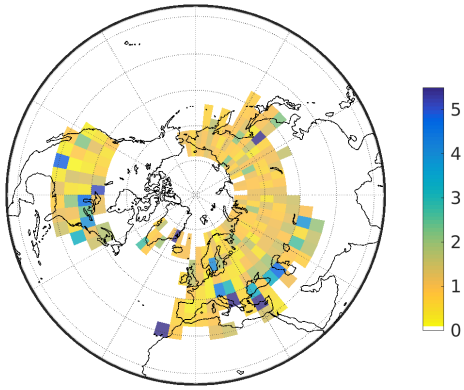
(a) *σ for the model EOF, where $\sigma_{model} = 0.2 \cdot \lambda_1 = 4.9$ (red) and σ_{EOF} (blue) are shown separately. The peak on the right comes from the fact that the second model EOF has high values near Greenland.*



(b) *$\widetilde{\sigma}$ for the observation EOF, where $\widetilde{\sigma}_{meas}$ (red) and $\widetilde{\sigma}_{EOF}$ (blue) are shown separately. Missing values of the red line are actually infinite, since there are no measurements for those positions.*



(c) *σ for the model EOF*



(d) *$\widetilde{\sigma}$ for the observation EOF.*

Figure 30: *Values of the standard deviation (square root of variance) for the model EOF (left) and the observation EOF (right). These values are in °C, multiplied by $\lambda_1 = 24.53$, the first eigenvalue.*

num. years avg.	1	2	5	10	20
n	149	74	29	14	7
$\widetilde{n}^*$	41	21	8	4	2

Table 5: *Estimate of effective sample size $\widetilde{n}^*$ for observations, along with the number of time periods n, for different averaging periods. $\widetilde{n}^*$ is averaged to the nearest integer.*

we don't take into account the DINEOF algorithm, we do account for the partial coverage of the ECA and USHCN datasets.

Just like in the previous section, we assume the two sources of error to be independent. This allows us to calculate the total variance $\widetilde{\sigma}_i^2$ by adding the variance from the measurements with the variance from the EOFs:

$$\widetilde{\sigma}_i^2 = \widetilde{\sigma}_{i,\text{meas}}^2 + \widetilde{\sigma}_{i,\text{EOF}}^2.$$

This is plotted in Fig. 30b and 30d.

The error covariance matrix of the observations EOF, $\mathbf{R'}_m$, is now constructed in the same way as in section 9.3.1, using the variances just defined.

9.4 Kalman filter data assimilation

Now that we have defined the error covariance matrices for the EOFs, we can look at the data assimilation process. We rewrite the Kalman filter equations to obtain:

$$\begin{aligned} \mathbf{e}^a &= \mathbf{e}^b + \mathbf{K'}(\mathbf{e}^o - \mathbf{H'}\mathbf{e}^b), \\ \text{with } \mathbf{K'} &= \mathbf{P'}^b(\mathbf{H'})^T \left(\mathbf{H'}\mathbf{P'}^b\mathbf{H'}^T + \mathbf{R'}\right)^{-1} \end{aligned} \tag{34}$$

Here, $\mathbf{e}^a$ is the assimilated EOF, $\mathbf{e}^o$ the EOF of the observations and $\mathbf{e}^b$ the background EOF, which is the EOF of the model anomalies. We also use the

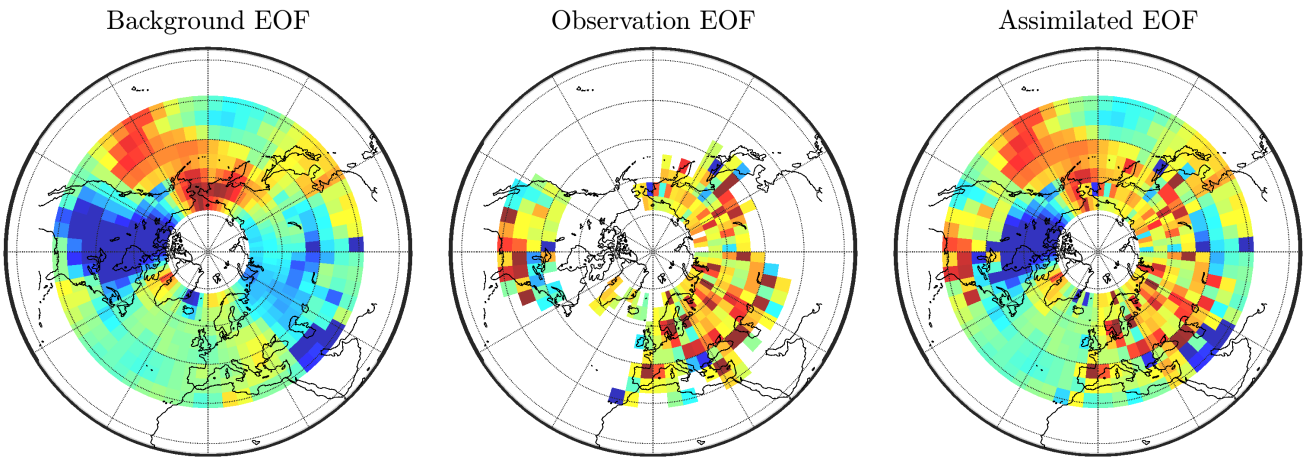


Figure 31: *Results of data assimilation for 5 year average. The background EOF (left) is combined with the observation EOF (middle), to obtain the assimilated EOF (right). With this standard deviation, the Kalman gain uses values of the observation EOF when they are available (Europe, USA, etc.), and uses the model EOF for the rest (Pacific Ocean, Canada, etc.).*

prime notation for the Kalman gain for EOFs, $\mathbf{K}'$ and the observation operator $\mathbf{H}'$.

For the same reason we only used the first EOF/PC in section 7, we only perform data assimilation using the first EOF of the model output and of the observations.

Before we can perform data assimilation, we need to define the observation operator on EOFs, $\mathbf{H}'$. This operator maps the background field to the observation space. In this case, the observation operator maps the model anomalies EOF to the raw observations from the ECA/USHCN measurement stations. It is therefore an "inverse EOF decomposition". This is impossible to achieve exactly, since one EOF doesn't contain enough information to obtain the original anomalies from it, which can be interpolated to the measurement stations. For this reason, we consider the observation field to be the EOF of the interpolated, averaged measurement anomalies, and we keep in mind that the Kalman gain is just an approximation.

Since some grid points don't have any measurement at all for the whole period 1850 to 1998, the value of the EOF at these points should be ignored. For this reason, we limit the observation space to all the grid points having at least one measurement. Instead of having $64 \times 8 = 512$ grid points, the observation space now has 241 grid points. We define the observation operator $\mathbf{H}'$ exactly as in section 8.6. Let $\{i_1^o, i_2^o, ..., i_2^o 41\}$ be the set of indices of the grid points with at least one measurement. The observation operator $\mathbf{H}'$ is a 241×512 matrix, where the j-th row is made of all zeros, except for the i_j^o-th column, which is 1. An example of this is also given in section 8.6.

When applying the data assimilation process using this observation operator, for 5 year averages and using a standard deviation of the model of $\sigma = 0.2$, we obtain the EOF shown in Fig. 31. The right figure shows the assimilated EOF. It is indeed a combination of the background EOF and the observation EOF.

9.5 Reconstruction of assimilated field

Since the data assimilation process is calculated directly on the EOFs, we still have to reconstruct the anomalies from the first assimilated EOF and the original EOFs. By definition, we know that the background field can be reconstructed using:

$$X_{\text{model}} = \sum_{m=1}^{n_{\text{EOF}}} A_{\text{model},m} (\mathbf{e}_m^b)^T. \tag{35}$$

In this equation, $A_{\text{model},m}$ is the m-th principal component of the model anomalies, and n_{EOF} is the total number of EOFs. This is equal to $n_{\text{EOF}} = \min(p, n)$ with p the number of grid points and n the number of time periods. We can transform Eq. (35) to obtain the reconstructed field, where we use as first EOF the assimilated EOF. The other EOFs are left untouched: we use the model EOFs for them. The reconstructed

field $X_{\text{reconstr.}}$ is equal to:

$$X_{\text{reconstr.}} = A_{\text{model},1}(\mathbf{e}^a)^T + \sum_{m=2}^{n_{\text{EOF}}} A_{\text{model},m}(\mathbf{e}^b_m)^T,$$

where $\mathbf{e}^a$ is the assimilated EOF.

The last step is to compare the reconstructed field to the HadCRUT measurements. This is discussed in the next section.

9.6 Quality assessment with HadCRUT

In section 8.8, we argue that the highest correlations are obtained when projecting the model anomalies onto the EOF of the HadCRUT dataset. We do exactly the same here: we project the reconstructed assimilated field onto the EOFs of the HadCRUT dataset, and we compare the resulting first PC to the first HadCRUT PC.

The comparison is performed using the same two measures of skill we have already used before: the correlation coefficient between the first PCs, and the RMS difference between the two. We do the same for the non-assimilated background field (the original model EOF), and compare the correlation and RMS difference to the assimilated ones.

The first principal components of the projected field using the assimilated EOFs (blue) compared to the first PC of the HadCRUT dataset (red) are shown in Fig. 32a to 32d. As a reference, we also include the non-assimilated background PC in gray. While the changes enduced by the data assimilation are small, the first PC is indeed affected by it. The fact that the changes are small can be explained by the fact that we only assimilate the first EOF. A more quantitative method of describing the changes obtained through the data assimilation, is by looking at the correlation coefficients and RMS differences between the assimilated PC and the HadCRUT PC. These are shown in Fig. 32e and 32f. Indeed, the changes between assimilated and original model are small, but for the 1 year and 20 year averages, there is an increase in correlation. The RMS errors are smaller for the assimilated EOF for smaller time scales (1 and 2 years), and larger than the non-assimilated EOF for time scales of 5 years or longer.

This shows us that, even though we have made many assumptions, there is still some improvement in the assimilated field, albeit very small. For time scales of 2, 5 and 10 years, the assimilated field is actually less similar to the HadCRUT data.

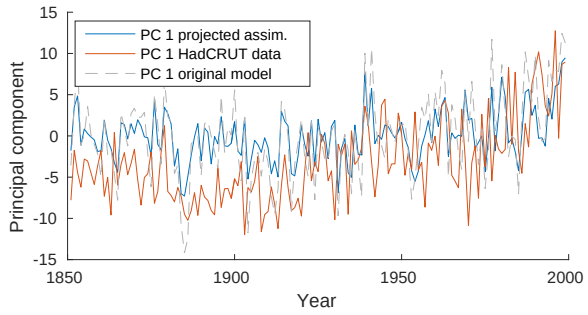
10 Summary and conclusions

The goal of this thesis was twofold: to implement various methods to assess the errors of a climate model, and to reduce these errors using observations.

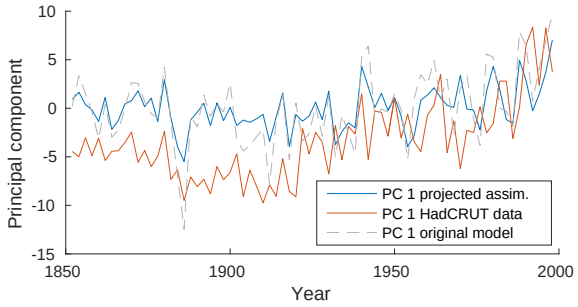
A first measure of the errors of the LOVECLIM model we considered was to average the model output and the HadCRUT dataset over a specific area, and compare the obtained time series. We used the correlation coefficient and RMS errors between the time series to quantify their differences. In general, the model performs better for longer time scales. This can be expected, since LOVECLIM is a climate model for the past on relatively long time scales. By averaging over Europe, the model is in agreement with the data for the last 50 years, but has bigger discrepancies for the years before 1930. The average over the full Northern Hemisphere gives exactly the opposite results: there are large differences for the last 50 years. This already shows that the area considered when averaging has a large influence on the results. This method was therefore not enough to properly assess the quality of the temperature component of the model.

A more advanced method is to look at the patterns of highest variability, instead of averaging over arbitrary spatial areas. This was done by decomposing the anomaly sets into temporary patterns (principal components) and spatial patterns (EOFs). We then projected one set onto the first EOF of the other, giving two principal components. For monthly averages, the correlation between these PCs is practically zero. The longer the time scale, the higher the correlation. The Southern Hemisphere yields very big differences in correlation coefficients when projecting the data on the model compared to when projecting the model on the data. These results, however, are unreliable, since the Southern Hemisphere lacks many observations. Moreover, we showed that only the first PC/EOF could be used for this analysis: the model anomalies have hardly any variation along the second EOF of the data.

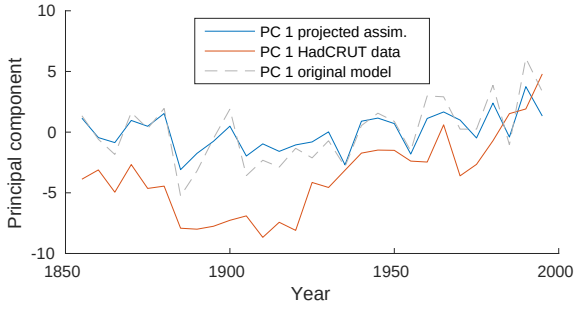
The third method consists of employing Kalman filter data assimilation to estimate the model error.



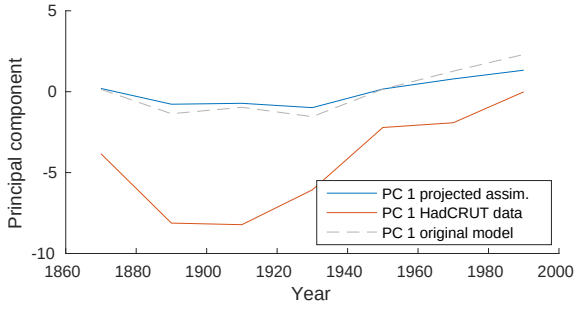
(a) *First principal components, yearly average*



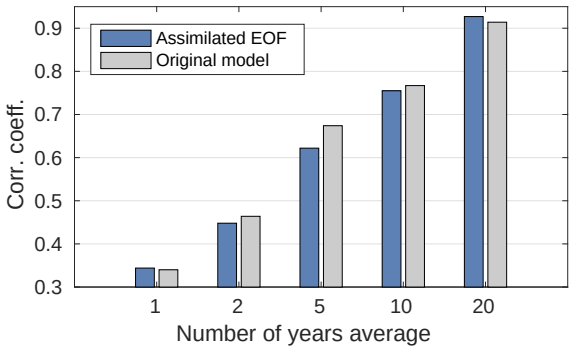
(b) *First principal components, 2 year average*



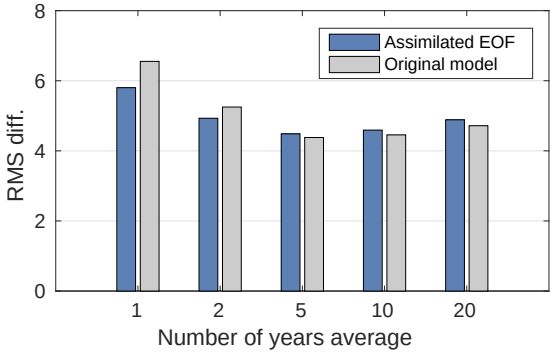
(c) *First principal components, 5 year average*



(d) *First principal components, 20 year average*



(e) *Correlation coefficients as function of averaging period*



(f) *RMS differences as function of averaging period*

Figure 32: First principal component *of the projected assimilated field and the HadCRUT dataset for different averaging periods. The blue line is the PC of the assimilated field using data assimilation on EOFs, the gray line is the PC of the model output without data assimilation. The correlation coefficients and RMS differences between the projected assimilated PC and the HadCRUT PC are shown in (e) and (f).*

We performed data assimilation between the model anomalies, having *unknown* variance, and the observations from the ECA/USHCN dataset, having a *given* variance, by chosing different values of the model variance. The assimilated field was then compared to the HadCRUT dataset, which we considered to be the reference dataset, by using the same EOF/PC decomposition as the second method. It appears that for a standard deviation of $0.2°C$, the correlation coefficient is the highest and the RMS error is the lowest between the first principal component of the assimilated field and of the HadCRUT dataset.

Moreover, the Kalman filter provides with an estimation of the anomalies that better resembles the HadCRUT observations than the anomalies from the LOVECLIM model alone, which indicates a reduction in the model error.

The last part of this thesis introduces the possibility to perform data assimilation directly on EOFs. This is a new method, which, as to the day of writing, has never been applied before. We have shown that, while this method brings a number of difficulties, it is indeed possible. First, the variance of every grid point is changed because of an additional error term coming from the EOF decomposition itself. We used the rule of thumb by North et al. to estimate this term. The second difficulty arises when using a spatial domain containing grid points without any measurements. This can be dealt with by limiting the observation space to the grid points having at least one measurement, and using an approriate observation operator. If we want to perform more effective and accurate data assimilation on the EOFs, we will have to elaborate the method, by looking more carefully at the many assumptions we had to make (like the form of the covariance matrix, of the observation operator, the number of EOFs considered and the Kalman filter itself). It could also be possible to extend this to different data assimilation schemes, such as ensemble data assimilation. This is left for further research.

It should be underlined that the methods developed and applied during this thesis to the LOVECLIM model and the HadCRUT dataset are not limited to these, and therefore can be applied to any model or dataset that is time and space-dependent.

References

Accadia, C., Mariani, S., Casaioli, M., Lavagnini, A. & Speranza, A. Sensitivity of Precipitation Forecast Skill Scores to Bilinear Interpolation and a Simple Nearest-Neighbor Average Method on High-Resolution Verification Grids. *Weather and Forecasting*, **18**:918–932 (2003). doi:10.1175/1520-0434(2003)018<0918:SOPFSS>2.0.CO;2.

Baldwin, M.P., Stephenson, D.B. & Jolliffe, I.T. Spatial weighting and iterative projection methods for EOFs. *Journal of Climate*, **22** (2):234–243 (2009). doi:10.1175/2008JCLI2147.1.

Beckers, J.M. & Rixen, M. EOF calculations and data filling from incomplete oceanographic datasets. *Journal of Atmospheric and Oceanic Technology*, **20** (12):1839–1856 (2003). doi:10.1175/1520-0426(2003)020<1839:ECADFF>2.0.CO;2.

Bergthórsson, P. & Döös, B.R. Numerical Weather Map Analysis. *Tellus*, **7** (3):329–340 (1955). doi:10.1111/j.2153-3490.1955.tb01170.x.

Björnsson, H. *A Manual for EOF and SVD Analyses of Climatic Data* (1997).

Brohan, P., Kennedy, J.J., Harris, I., Tett, S.F.B. & Jones, P.D. Uncertainty estimates in regional and global observed temperature changes: A new data set from 1850. *Journal of Geophysical Research: Atmospheres*, **111** (12):1–21 (2006). doi:10.1029/2005JD006548.

Courtier, P., Andersson, E., Heckley, W., Pailleux, J., Vsiljevic, D., Hamrud, M., Hollingsworth, A., Rabier, F. & Fisher, M. The ECMWF implementation of three-dimensional variational assimilation (3D-Var). I: Formulation. *Q. J. R. Meteorol. Soc.*, **124** (1998):1783–1807 (1998). doi:10.1256/qj.04.67.

Cubasch, U., Wuebbles, D., Chen, D., Facchini, M.C., Frame, D., Mahowald, N. & Winther, J.G. Climate Change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change. *Cambridge University Press*, pages 119—-158 (2013).

Eliassen, A. Provisional Report on Calculation of Spatial Covariance and Autocorrelation of the Pressure

Field. *Rept. No. 5, Institute of Weather and Climate Res., Academy of Science Oslo*, page 11 (1954).

Eyre, J.R. Inversion of cloudy satellite sounding radiances by nonlinear optimal estimation. I: Theory and simulation for TOVS. *Q. J. R. Meteorol. Soc.*, **115** (489):1001–1026 (1989). doi:10.1256/qj.04.67.

Gandin, L. Objective Analysis of Meteorological Fields. *Gidrometeorol. Izd., Leningrad (in Russian); English translation by Israel Program for Scientific Translations, Jerusalem, 1965* (1963).

Gaspari, G., Cohn, S.E., Guo, J. & Pawson, S. Construction and application of covariance functions with variable length-fields. *Qjrms*, **132** (619):1815–1838 (2006). doi:10.1256/qj.05.08.

Ghil, M. Meteorological data assimilation for oceanographers. Part I: Description and theorectical framework (1989).

Gilchrist, B. & Cressman, G.P. An Experiment in Objective Analysis. *Tellus*, **6** (4):309–318 (1954). doi:10.1111/j.2153-3490.1954.tb01126.x.

Golub, G.H. & Reinsch, C. Singular value decomposition and least squares solutions. *Numerische Mathematik*, **14** (5):403–420 (1970). doi:10.1007/BF02163027.

Goosse, H., Brovkin, V., Fichefet, T., Haarsma, R., Huybrechts, P., Jongma, J., Mouchet, A., Selten, F., Barriat, P.Y., Campin, J.M., Deleersnijder, E., Driesschaert, E., Goelzer, H., Janssens, I., Loutre, M.F., Morales Maqueda, M.A., Opsteegh, T., Mathieu, P.P., Munhoven, G., Pettersson, E.J., Renssen, H., Roche, D.M., Schaeffer, M., Tartinville, B., Timmermann, A. & Weber, S.L. Description of the Earth system model of intermediate complexity LOVECLIM version 1.2. *Geoscientific Model Development*, **3** (2):603–633 (2010). doi:10.5194/gmd-3-603-2010.

Hannachi, A., Jolliffe, I.T. & Stephenson, D.B. Empirical orthogonal functions and related techniques in atmospheric science: A review. *International Journal of Climatology*, **27** (May):1119–1152 (2007). doi:10.1002/joc.

Higham, N.J. Computing a nearest symmetric positive semidefinite matrix. *Linear algebra and its applications*, **103**:103—-118 (1988).

Kalman, R. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, **82** (1):35–45 (1960). doi:10.1115/1.3662552.

Klein Tank, A.M.G., Wijngaard, J.B., Können, G.P., Böhm, R., Demarée, G., Gocheva, A., Mileta, M., Pashiardis, S., Hejkrlik, L., Kern-Hansen, C., Heino, R., Bessemoulin, P., Müller-Westermeier, G., Tzanakou, M., Szalai, S., Pálsdóttir, T., Fitzgerald, D., Rubin, S., Capaldo, M., Maugeri, M., Leitass, A., Bukantis, A., Aberfeld, R., Van Engelen, A.F.V., Forland, E., Mietus, M., Coelho, F., Mares, C., Razuvaev, V., Nieplova, E., Cegnar, T., Antonio López, J., Dahlström, B., Moberg, A., Kirchhofer, W., Ceylan, A., Pachaliuk, O., Alexander, L.V. & Petrovic, P. Daily dataset of 20th-century surface air temperature and precipitation series for the European Climate Assessment. *International Journal of Climatology*, **22** (12):1441–1453 (2002). doi:10.1002/joc.773.

Knutti, R. & Hegerl, G.C. The equilibrium sensitivity of the Earth's temperature to radiation changes. *Nature Geoscience*, **1** (11):735–743 (2008). doi:10.1038/ngeo337.

Kutzbach, J.E. Empirical Eigenvectors of Sea-Level Pressure, Surface Temperature and Precipitation Complexes over North America (1967). doi:10.1175/1520-0450(1967)006<0791:EEOSLP>2.0.CO;2.

Lambers, M. Mappings between Sphere , Disc , and Square. **5** (2) (2016).

Lloyd, J. & Taylor, J.A. On the Temperature Dependence of Soil Respiration. *Functional Ecology*, **8**:315–323 (1994).

Lorenz, E.N. Empirical Orthogonal Functions and Statistical Weather Prediction (1956).

Menne, M.J., Williams, C.N. & Vose, R.S. The U.S. historical climatology network monthly temperature data, version 2. *Bulletin of the American Meteorological Society*, **90** (7):993–1007 (2009). doi:10.1175/2008BAMS2613.1.

North, G.R., Bell, T.L. & Cahalan, R.F. Sampling Errors in the Estimation of Empirical Orthogonal Funtions (1982). doi:10.1175/1520-0493(1982)110<0699:SEITEO>2.0.CO;2.

Obukhov, A. Statistically homogeneous fields on a sphere. *Uspekhi Matematicheskikh Nauk*, **2**:196—-198 (1947).

Rijkswaterstaat. IJsverslag winter 1940-1941 (1942).

Taylor, M.H., Losch, M., Wenzel, M. & Schröter, J. On the Sensitivity of Field Reconstruction and Prediction Using Empirical Orthogonal Functions Derived from Gappy Data. *Journal of Climate*, **26** (22):9194–9205 (2013). doi:10.1175/JCLI-D-13-00089.1.

Thepaut, J.N. & Courtier, P. Four-dimensional variational data assimilation using the adjoint of a multilevel primitive-equation model. *Quarterly Journal of the Royal Meteorological Society*, **117** (502):1225–1254 (1991). doi:10.1002/qj.49711750206.

Thiébaux, H.J. & Zwiers, F.W. The Interpretation and Estimation of Effective Sample Size. *Journal of Climate and Applied Meteorology*, **23**:800—-811 (1984).

Appendices

A Data assimilation: derivation of analyzed error covariance matrix

Theorem 1. *In section 8.3.2, we used the fact that the error covariance matrix of the analyzed, $\mathbf{P}_k^a$, can be rewritten as:*

$$\mathbf{P}_k^a = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_k^b(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)^T + \mathbf{K}_k\mathbf{R}_k\mathbf{K}_k.$$

Proof. Starting with the definition of the error covariance matrix, we have:

$$\mathbf{P}_k^a = \mathrm{E}[(\mathbf{w}_k^a - \mathbf{w}_k^t)(\mathbf{w}_k^a - \mathbf{w}_k^t)^T].$$

For clarity reasons, we rewrite this as:

$$\mathbf{P}_k^a = \mathrm{E}[\mathbf{w}_k^a - \mathbf{w}_k^t]^2,$$

keeping in mind that we are working with vectors, not scalars.

Using the definition of the analyzed field (Eq. (16)), and rewriting the expression, we obtain successively:

$$\begin{aligned}
\mathbf{P}_k^a &= \mathrm{E}[\mathbf{w}_k^b + \mathbf{K}_k(\mathbf{w}_k^o - \mathbf{H}_k\mathbf{w}_k^b) - \mathbf{w}_k^t]^2 \\
&= \mathrm{E}[(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{w}_k^b + \mathbf{K}_k\mathbf{w}_k^o - \mathbf{w}_k^t]^2 \\
&= \mathrm{E}[(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{w}_k^b + \mathbf{K}_k\mathbf{w}_k^o - \mathbf{w}_k^t \\
&\qquad + \mathbf{K}_k\mathbf{H}_k\mathbf{w}_k^t - \mathbf{K}_k\mathbf{H}_k\mathbf{w}_k^t]^2 \\
&= \mathrm{E}[(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)(\mathbf{w}_k^b - \mathbf{w}_k^t) + \mathbf{K}_k\mathbf{w}_k^o - \mathbf{K}_k\mathbf{H}_k\mathbf{w}_k^t]^2.
\end{aligned}$$

Since we know the formulation of the observation field (Eq. (17), $\mathbf{w}_k^o = \mathbf{H}_k\mathbf{w}_k^t + \mathbf{b}_k^o$), we find:

$$\begin{aligned}
\Leftrightarrow\ \mathbf{P}_k^a &= \mathrm{E}[(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)(\mathbf{w}_k^b - \mathbf{w}_k^t) \\
&\qquad + \mathbf{K}_k\mathbf{H}_k\mathbf{w}_k^t + \mathbf{K}_k\mathbf{b}_k^o - \mathbf{K}_k\mathbf{H}_k\mathbf{w}_k^t]^2 \\
&= \mathrm{E}[(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)(\mathbf{w}_k^b - \mathbf{w}_k^t) + \mathbf{K}_k\mathbf{b}_k^o]^2
\end{aligned}$$

By remembering the notation $\mathrm{E}[..]^2 := \mathrm{E}[(..)(..)^T]$, we obtain the long form:

$$\begin{aligned}
\Leftrightarrow\ \mathbf{P}_k^a &= \mathrm{E}[((\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)(\mathbf{w}_k^b - \mathbf{w}_k^t) + \mathbf{K}_k\mathbf{b}_k^o) \\
&\qquad ((\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)(\mathbf{w}_k^b - \mathbf{w}_k^t) + \mathbf{K}_k\mathbf{b}_k^o)^T] \\
&= \mathrm{E}[((\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)(\mathbf{w}_k^b - \mathbf{w}_k^t) + \mathbf{K}_k\mathbf{b}_k^o) \\
&\qquad ((\mathbf{w}_k^b - \mathbf{w}_k^t)^T(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)^T + (\mathbf{b}_k^o)^T\mathbf{K}_k^T)] \\
&= (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\,\mathrm{E}[(\mathbf{w}_k^b - \mathbf{w}_k^t)(\mathbf{w}_k^b - \mathbf{w}_k^t)^T](\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)^T \\
&\qquad + \mathbf{K}_k\,\mathrm{E}[\mathbf{b}_k^o(\mathbf{b}_k^o)^T]\mathbf{K}_k^T \\
&\qquad + \text{cross terms in } \mathrm{E}[\mathbf{b}_k^o]
\end{aligned}$$

Since the expectation value of $\mathbf{b}_k^o$ is zero, the cross terms in $\mathrm{E}[\mathbf{b}_k^o]$ become zero. Therefore, we are left with the two first terms only. We recognize the definition of $\mathbf{P}_k^b$ (Eq. (18)) and $\mathbf{R}_k$:

$$\begin{aligned}
\mathbf{P}_k^b &:= \mathrm{E}[(\mathbf{w}_k^b - \mathbf{w}_k^t)(\mathbf{w}_k^b - \mathbf{w}_k^t)^T], \\
\mathbf{R}_k &:= \mathrm{E}[\mathbf{b}_k^o(\mathbf{b}_k^o)^T].
\end{aligned}$$

Substituting in these definitions, we obtain the final answer:

$$\Leftrightarrow\ \mathbf{P}_k^a = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_k^b(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)^T + \mathbf{K}_k\mathbf{R}_k\mathbf{K}_k,$$

which concludes our proof. □

B Data assimilation: derivation of Kalman filter

Theorem 2. *When performing Kalman filter data assimilation, we need an expression for the Kalman filter, or*

gain factor. For a linear model and linear observation operator $\mathbf{H}_k$, the Kalman filter $\mathbf{K}_k = \mathbf{K}_k^*$ that minimizes the analyzed error variance is given by:

$$\mathbf{K}_k^* = \mathbf{P}_k^b \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^b \mathbf{H}_k^T + \mathbf{R}_k)^{-1}.$$

Proof. The goal is to find a Kalman weight that minimizes the analyzed error variance:

$$J = \text{tr}(\mathbf{P}_k^a).$$

From Eq. (21), this is equal to:

$$\Leftrightarrow \; J = \text{tr}\left((\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^b (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k\right).$$

Minimizing this with respect to $\mathbf{K}_k$ means solving the equation

$$\begin{aligned} &\frac{\partial}{\partial \mathbf{K}_k}\left(\text{tr}(\mathbf{P}_k^a)\right) = 0 \\ \Leftrightarrow \; &\frac{\partial}{\partial \mathbf{K}_k}(\text{tr}((\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^b (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T \\ &\qquad + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k)) = 0. \end{aligned}$$

We solve this using the following matrix calculus identity. If A and B are two matrices, with B symmetric, the derivative of ABA^T with respect to A is equal to:

$$\frac{\partial}{\partial A}\,\text{tr}(ABA^T) = 2AB.$$

Since $\mathbf{P}_k^b$ and $\mathbf{R}_k$ are covariance matrices, they are symmetric. Hence, this identity can be applied here:

$$\begin{aligned} &\frac{\partial}{\partial \mathbf{K}_k}(\text{tr}((\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^b (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k)) \\ &= \frac{\partial}{\partial \mathbf{K}_k}(\text{tr}((\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^b (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T)) \\ &\qquad + \frac{\partial}{\partial \mathbf{K}_k}(\text{tr}(\mathbf{K}_k \mathbf{R}_k \mathbf{K}_k)) \\ &= \frac{\partial}{\partial \mathbf{K}_k}(\text{tr}((\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^b (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T)) + 2\mathbf{K}_k \mathbf{R}_k. \end{aligned}$$

The second step can be done because of the linearity of the derivative and the trace operator. Now we still have a problem with the first term:

$$\frac{\partial}{\partial \mathbf{K}_k}(\text{tr}((\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^b (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T)),$$

since we are deriving to $\mathbf{K}_k$, but we have a term $(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^b (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T)$, and not $\mathbf{K}_k \mathbf{P}_k^b \mathbf{K}_k^T$. We use the chain rule, and differentiate with respect to $(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)$. In matrix calculus, this leads to:

$$\begin{aligned} &\frac{\partial}{\partial \mathbf{K}_k}(\text{tr}((\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^b (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T)) \\ &= \frac{\partial}{\partial (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)}(\text{tr}((\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^b (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T))(-\mathbf{H}_k^T) \\ &= -2(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^b \mathbf{H}_k^T. \end{aligned}$$

Bringing all the terms together, gives us:

$$\begin{aligned} &\frac{\partial}{\partial \mathbf{K}_k}\left(\text{tr}(\mathbf{P}_k^a)\right) = 0 \\ \Leftrightarrow \; &-2(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^b \mathbf{H}_k^T + 2\mathbf{K}_k \mathbf{R}_k = 0 \\ \Leftrightarrow \; &-\mathbf{P}_k^b \mathbf{H}_k^T + \mathbf{K}_k \mathbf{H}_k \mathbf{P}_k^b \mathbf{H}_k^T + \mathbf{K}_k \mathbf{R}_k = 0 \\ \Leftrightarrow \; &\mathbf{K}_k(\mathbf{H}_k \mathbf{P}_k^b \mathbf{H}_k^T + \mathbf{R}_k) = \mathbf{P}_k^b \mathbf{H}_k^T \\ \Leftrightarrow \; &\mathbf{K}_k = \mathbf{P}_k^b \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^b \mathbf{H}_k^T + \mathbf{R}_k)^{-1}, \end{aligned}$$

which is the final form of the Kalman filter. □