Exercise 4 – Does the sun ever shine in Denmark

Mathematical Modelling
02526

The goal of this exercise is to use mathematical modeling to solve a practical problem: Predicting the electricity production from solar cells. You should:

- Identify significant issues
- Formulate mathematical models, which solve these issues
- Implement the model in a computer program
- Use the implemented model for the analysis of data
- Report the results from the analysis and discuss the validity of the model

The report for this exercise must be 5 pages at most, including graphs, tables, and images (excluding the front page and appendices). In this exercise, several questions are asked. The report, however, should not be a list of answers, but instead be a coherent documentation and discussion of the analysis performed.

The exercise consists of several parts. The first is understanding the problem with solar cells in the electrical grid, and the nature of satellite imaging. The next is to understand what is necessary for a regression algorithm, and how we can use satellite imaging for such an algorithm. Finally, you will formulate and implement a prediction algorithm that can help the grid operator manage the electricity grid with increasing amounts of solar power.

To solve the exercise, we suggest you use PYTHON.

Predicting electricity production from solar cells

The national grid in Denmark is managed by a publicly owned company EnergiNet. They need to balance the grid, as production and demand *must constantly* be matched. If there is too much production the voltage spikes and can damage equipment - both in the net and at the consumers. If there is too little production the voltage drops, and if it gets too low production units shut down as a safety measure and we have a blackout.

The problem is, that the output from solar cells is highly variable. If clouds pass a solar farm the production can go from nearly full capacity to a small fraction in a matter of minutes if not seconds. See fig. 1 for an example. This can be managed to some degree with reserves that can quickly produce more, but they are expensive to maintain and the cost is ultimately sent to the consumers.

If the output can be predicted more accurately EnergiNet can manage a larger fraction of solar power more efficiently and thus both cheaper and safer for grid stability.

That means we want a predictive algorithm that can tell EnergiNet what the solar production will be in X minutes into the future, such that they can scale other electricity sources up or down as needed. Further, the algorithm should be transparent, i.e. interpretable. For something as critical as the power grid, we do not want any "black box" models.

Satellite imaging

The data we have available are from the Meteosat programme. They are meteorological satellites in geosynchronous orbit, i.e. they maintain their position relative to the surface of the earth and take images every 15 minutes. So-called High Rate SEVIRI Level 1.5 Image Data. An example of the view in so-called "natural colors" can be seen in fig. 2

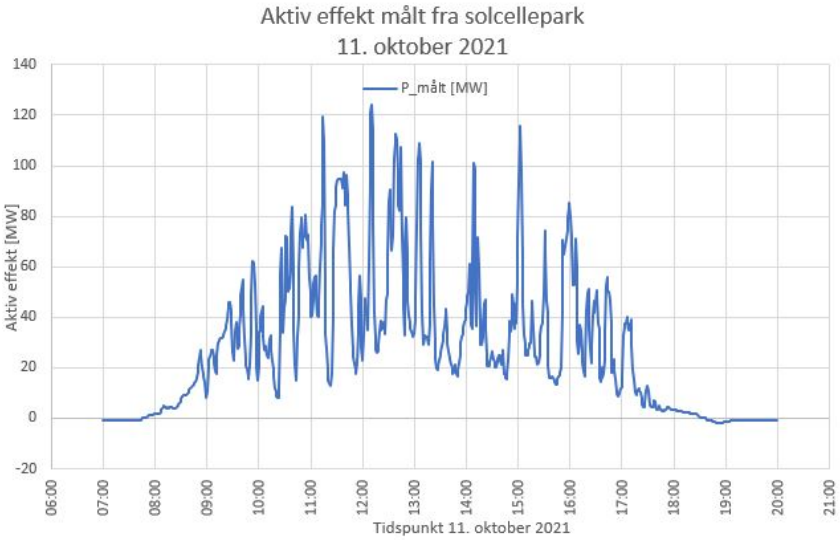

Figure 1: The production curve for a solar cell plant for a single day. The impact of clouds is clearly seen from the ragged shape of the production curve. EnergiNet

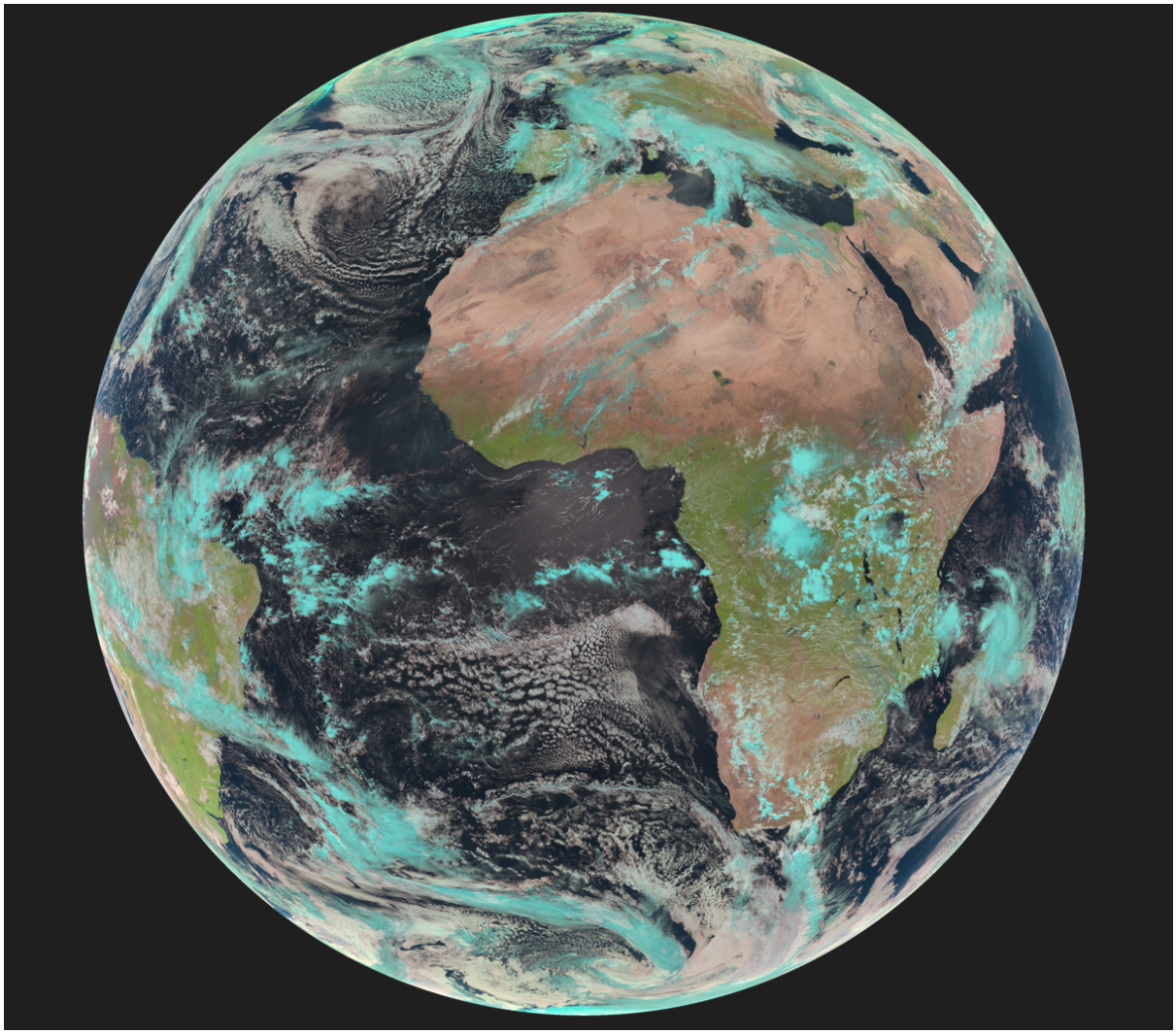

Figure 2: The earth on the 26th of March 12:12 o'clock CET. Note how small a part Denmark is of the image. source

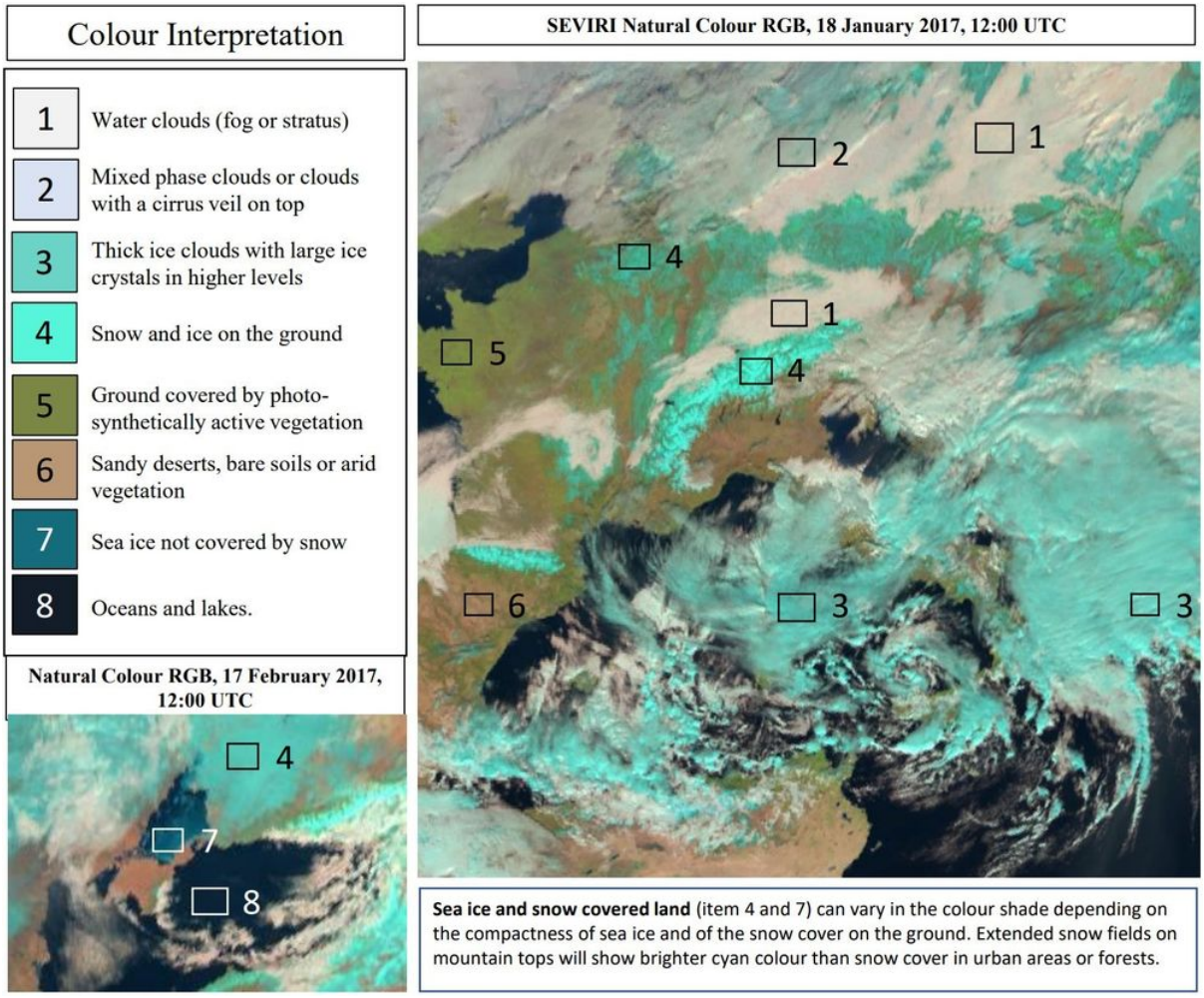

Figure 3: Guide to the natural color satellite images. Source.

The satellite records in 10 bands however, we will only use three of them, the red green, and blue: "*The Natural Colour RGB (Red, Green, Blue) makes use of three solar channels: NIR1.6, VIS0.8 and VIS0.6. In this color scheme vegetation appears greenish because of its large reflectance in the VIS0.8 channel (the green beam) compared to the NIR1.6 (red beam) and VIS0.6 (blue beam) channels. Water clouds with small droplets have large reflectance at all three channels and hence appear whitish, while snow and ice clouds appear cyan because ice strongly absorbs in NIR1.6 (no red). Bare ground appears brown because of the larger reflectance in the NIR1.6 than at VIS0.6, and the ocean appears black because of the low reflectance in all three channels.*" from eumetsat.int. Further explanation can be found here. A guide is shown in fig. 3.

We can easily see the clouds. However, it is where there are no clouds we will have the largest production. We can make a new image based on the three channels Red Green and Blue (RGB). We can see that the clouds are either white or bluish while the land is brown and green. We create a new image by $NoCloud = R + G - 2 \cdot B$. An example is see in fig. 4 and 5

Solar production curves Solar power follows a predictable pattern determined by the height of the sun - as long as there are no clouds! We can access actual production data from EnergiNet to do curve fitting, or it can be based on theoretical models.

By the end of last year, there was an installed capacity of 3529 MW[1]. This is the theoretical max

[1] Energistyrelsen

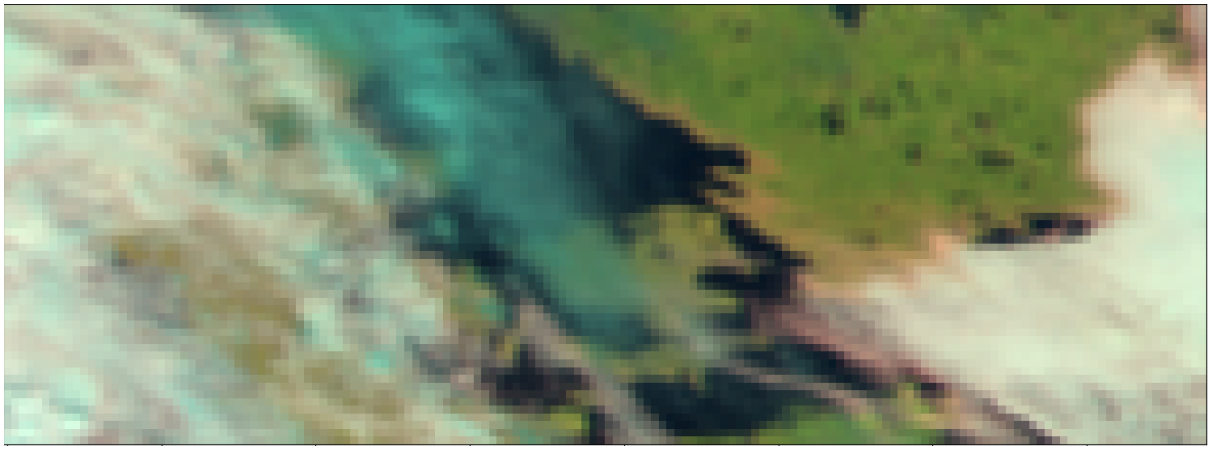

Figure 4: A natural-color image of Denmark. We see parts of southern Jutland, Zealand, and Sweden are cloud-free.

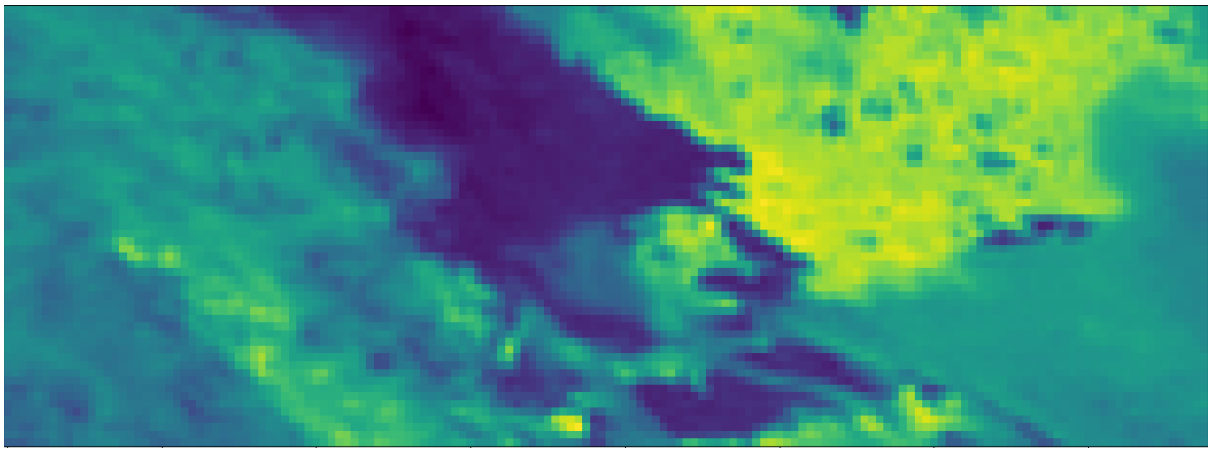

Figure 5: The image in fig. 4 transformed by $NoCloud = R + G - 2 \cdot B$

production is all installations get an optimal amount of sun and is thus an upper limit.

Regularised regression Since we have many more variables (pixels) than samples (production at the time of the image), we cannot use normal unconstrained regression. The following is an excerpt from [1].

In the analysis of regression models, one often will have stability problems if the design matrix is ill-conditioned. This may be detected by suitable regression diagnostics. If we have a model with many independent variables a solution to this problem may be various stepwise regression procedures. This will however not always work satisfactorily, so instead of excluding some variables and focusing completely on others, one might try to utilize the information available in all independent variables differently.
We consider the usual model

$$\boldsymbol{Y} = \mathbf{x}\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

where $\mathbf{x}$ is a known $n \times p$ matrix, $\boldsymbol{\beta}$ the unknown parameter vector and $\boldsymbol{\varepsilon}$ the error vector.

We assume that

$$\begin{aligned} \mathrm{E}(\boldsymbol{\varepsilon}) &= \mathbf{0} \\ \Delta L(\boldsymbol{\varepsilon}) &= \sigma^2 \mathbf{I}_n. \end{aligned}$$

The ordinary least squares estimator assumes that $\boldsymbol{x}$ has maximum rank -

$$\hat{\boldsymbol{\beta}} = (\mathbf{x}^T\mathbf{x})^{-1}\mathbf{x}^T\boldsymbol{Y}.$$

Furthermore, we assume that the independent variables are scaled so that $\mathbf{x}^T\mathbf{x}$ has correlation form (i.e. the single independent variables are reduced by their average and divided by their standard deviation). This normalization will help make the estimates more stable numerically. This normalization is often recommendable in a practical situation.

If $\mathbf{x}^T\mathbf{x}$ in this form is close to a unity matrix, i.e. if the independent variables are near-orthogonal, the least squares estimator is fine. If we have ***multicollinearity***, i.e. if the independent variables are strongly correlated, the estimates $\hat{\boldsymbol{\beta}}$ will be unstable.

We now analyze some properties of $\hat{\boldsymbol{\beta}}$ that has not been dealt with earlier. We have

$$\Delta L(\hat{\boldsymbol{\beta}}) = \sigma^2(\boldsymbol{x}^T\boldsymbol{x})^{-1}.$$

If we put L equal to the distance from $\hat{\boldsymbol{\beta}}$ to $\boldsymbol{\beta}$,

$$L^2 = (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^T(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}),$$

we get

$$\mathrm{E}(L^2) = \sum_{i=1}^{p} \mathrm{E}[(\hat{\beta}_i - \beta_i)^2] = \sum_{i=1}^{p} \mathrm{V}(\hat{\beta}_i) = \sigma^2(\boldsymbol{x}^T\boldsymbol{x})^{-1}.$$

Since

$$L^2 = \hat{\boldsymbol{\beta}}^T\hat{\boldsymbol{\beta}} - 2\hat{\boldsymbol{\beta}}^T\boldsymbol{\beta} + \boldsymbol{\beta}^T\boldsymbol{\beta},$$

we get that the expected value of the squared length of $\hat{\boldsymbol{\beta}}$ is

$$\mathrm{E}(\hat{\boldsymbol{\beta}}^T\hat{\boldsymbol{\beta}}) = \sigma^2(\mathbf{x}^T\mathbf{x})^{-1} + \boldsymbol{\beta}^T\boldsymbol{\beta}.$$

If we denote the eigenvalues for $\mathbf{x}^T\mathbf{x}$

$$\lambda_1 \geq \cdots \geq \lambda_p,$$

we obtain

$$\mathrm{E}(L^2) = \sigma^2(\frac{1}{\lambda_1} + \cdots + \frac{1}{\lambda_p}) > \frac{\sigma^2}{\lambda_p},$$

and

$$\mathrm{E}(\hat{\boldsymbol{\beta}}^T\hat{\boldsymbol{\beta}}) = \sigma^2(\frac{1}{\lambda_1} + \cdots + \frac{1}{\lambda_p}) + \boldsymbol{\beta}^T\boldsymbol{\beta} > \frac{\sigma^2}{\lambda_p} + \boldsymbol{\beta}^T\boldsymbol{\beta}.$$

If the independent variables are strongly correlated the eigenvalues of $\mathbf{x}^T\mathbf{x}$ will vary a lot and consequently the smallest will be very small ($<< 1$). According to the relations above the squared distance between $\boldsymbol{\beta}$ and $\hat{\boldsymbol{\beta}}$ will in this case have a large expected value, and the squared length of $\hat{\boldsymbol{\beta}}$ will have an expectation by far exceeding the squared length of $\boldsymbol{\beta}$.

This tendency to 'inflation' of $\hat{\boldsymbol{\beta}}$ is caused by requiring unbiasedness of $\boldsymbol{\beta}$. The question is therefore whether we by relaxing on this requirement may obtain estimates that in some sense are closer to $\boldsymbol{\beta}$.

Here we may again refer to the ***mean squared error*** of an estimator $\tilde{\beta}$ (in the one-dimensional case)

$$\mathrm{MSE}(\tilde{\beta}) = \mathrm{E}[(\tilde{\beta} - \beta)^2] = \mathrm{V}(\tilde{\beta}) + \{\mathrm{E}(\tilde{\beta}) - \beta\}^2,$$

i.e. that the MSE of an estimator is equal to the variance plus the squared bias. If we therefore by allowing a small bias may obtain a great reduction in the variance, this would obviously be preferable. This is exactly what is obtained with the ridge estimator introduced in the definition below.

A ***ridge estimator*** for β in the model

$$\boldsymbol{Y} = \mathbf{x}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

is an estimator $\hat{\boldsymbol{\beta}}_k^* = \hat{\boldsymbol{\beta}}^*$, that is a solution to

$$(\mathbf{x}^T\mathbf{x} + k \cdot \mathbf{I})\hat{\boldsymbol{\beta}}^* = \mathbf{x}^T\boldsymbol{Y},$$

i.e.

$$\hat{\boldsymbol{\beta}}^* = (\mathbf{x}^T\mathbf{x} + k \cdot \mathbf{I})^{-1}\mathbf{x}^T\boldsymbol{Y}.$$

Here k is a constant $\in [0, 1]$.

In numerical mathematics, such a way of solving normal equations is called a ***(Tikhonov) regularization***. This is a very common way of solving ill-posed problems.

Part 0 - data

You are given satellite data for 6 days in March for a total of 306 images together with production data in Excel. There is a script supplied to read the data. The images can be viewed in any picture viewer.

Inspect the satellite images and compare them to the production data. Does it make sense to use satellite images to estimate the production? Do you see a correlation? Use the mask supplied to extract the parts of the images that are part of Denmark.

Do you think it is better to use the natural color images or the modified images ($R + G - 2 \cdot B$) and why?

Make sure you can load both images and production curves and interact with them.

Note that more images will become available next week!

Part 1 - regression model

Make a regression model: How well can you predict the production using the satellite data? Do you need to use other data sources as well? Consider whether the production can be estimated directly from the satellite images, or whether we need to use them to estimate a deviation from a production curve. Can we fit a production curve to the data from EnergiNet or based on a model?

Note that there are other alternatives to regularisation than the Ridge introduced above. See e.g. The Elements of Statistical Learning. Whatever you use, you should understand it!

Part 2 - frame interpolation/extrapolation

Frame interpolation is simply to interpolate between two images with movement. One way is to estimate the optical flow and use that to estimate how the image should be moved between the two frames. This could be the global optical flow for the image if we have one weather system moving coherently.

You can use frame interpolation for at least two things. We can use it to generate more data points for the regression model, as we have the production for every minute. You can also use it to make a prediction algorithm: If we can extrapolate the image X minutes into the future, we can then simply apply the regression model and get an estimate for the production in the future, i.e. a prediction.

Part 3 - Prediction algorithm

The end goal of the project is to make a prediction algorithm. It is up to you what information you use. You can draw on other sources. e.g. meteorological observations such as temperature. You can use the movement of the sun, e.g. sunrise, sunset, and zenith. You can use time series analysis on the production data itself, and see if that is useful for prediction. Remember to think about validation and make a plan before you train on all the data.

The only constraint is, that your model should be interpretable. You should be able to explain how and why it works!

Reporting **Contents of the report**

1. Describe the problem and the background – what is modeled and why?
2. Describe data and experiments
3. Describe the mathematical model - how and why?
4. Describe results
5. Discuss results – how good are the results? To what degree do they reflect the true problem?
6. Conclude – what is the contribution of the analysis?

Hand in The report is uploaded to DTU Learn

References

1. Knut Conradsen, Anders Nymark Christensen, Allan Aasbjerg Nielsen, Bjarne Kjær Ersbøll *Multivariate Statistics for the Technical Sciences*, V0.94, Kgs. Lyngby, 2018.

Anders Nymark Christensen, Dimitrios Papadopoulos 2024