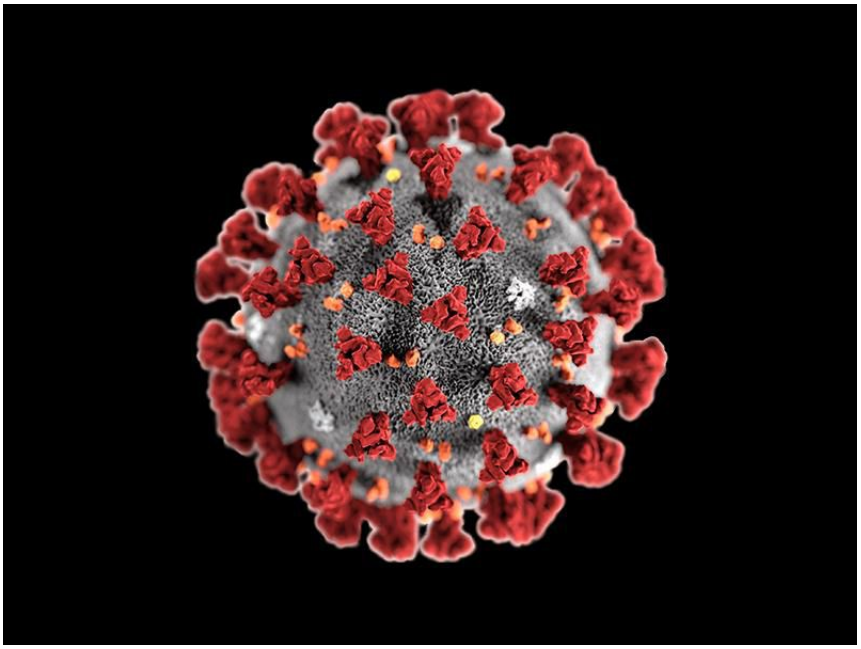
COVID-19-Modelling: Documentation

Gordon Pipa, Pascal Nieters, Daniel Rohe ....

April 2020



1 Introduction

In response to the covid-19 crisis we adopt the [1] to COVID-19 data from germany proviced by the RKI Data Link.

2 Goals

The presented approach has these goals.

- To estimate the effects of policy changes we will model exponential growth with a exponent that is a function of time

- To model the spread the spreading we will use spatial temporal kernels like in [1]
- We will mo dell new cases per day, and express the growth rate as number of days for doubling, and percentage change per day.
- We use a Bayesian approach to compute a posterior for the kernel and for the predictive distribution
- Potentially we want to explore effects by predicting future - see below several issues will come up than

3 Modelling Considerations

- Compared to [1] we have a single outbreak only
- Compared to [1] we have a apparent exponential growth with changing growth rate
- We have policy changes on the level of Landkreise, states, and on the level of Germany
- The Daily new cases have a periodic component with a weekly periodicity
- There is period of a few days at the end of the observed interval with an dip that is due to the delay of reported cases.
- The periodicity shall be modelled and than we removed to have a cleaned growth.

4 Working Hypotheses

We will construct 4 Working hypothesis. with the following goals.

- By comparing $WH1$, $WH2$ and $WH3$ we plan to show that beyond a certain initial growth period that all Lnadkreise can be modelled by the same space invariant kernel.
- As a next step we will use a single space invariant temporal kernel, scaled by the log population density and add the interaction effect- This interaction effect will be the key point of our work.

4.1 Working Hypothesis $WH1$

We assume that the shape and the speed of growth is identical for all Landkreise but scales by the log population density of the respective Landkreis. Therefore the temporal kernel is not dependent on the Landkreis or on space.

4.2 Working Hypothesis $WH2$

In contrast to $WH1$ the working hypothesis $WH2$ assumes that the temporal kernel is identical across all Landkreise, but scaled with a scaling parameter identified by the regression for each state. The difference between $WH1$ and $Wh2$ indicates that the log of the population is not sufficient to model the growth rate.

4.3 Working Hypothesis $WH3$

In contrast to $WH1$ the working hypothesis $WH3$ assumes that the temporal kernel is not identical across all Landkreise in shape and in scale. To restrict differences across the three Models, $WH1$, $WH2$ and $WH3$ we will build $WH3$ from $WH2$ by adding an addtional shape component. This additional shape component is the shape from $WH1$ raised by power of two. Therefore the Model $WH3$ models deviations from the scales shape of the temporal kernel from $WH2$.

4.4 Working Hypothesis $WH4$

The Working Hypothesis $WH4$ is build upon $Wk1$ and assumes that there is an spatial temporal interaction effect. it will be used as an additional component in the model of the Working Hypothesis $WH1$. For dense or less dense area the interaction kernel may absorb residuals for the growth rate that can not only explained by the log of the density.

- By comparing the two fits of the original and the alternative model we can assess whether and to what degree using the interaction kernel improved the model.
- By the shape of the interaction kernel we can assess the range of interaction and the speed of the spread.
- By the sensitivity analysis we can compare the importance of the interaction in contrast to the trend to check whether travel restrictions are indicated.

5 Model Implementation

5.1 Trend and Periodicity

We assume that the growth can modelled as a time dependent exponent $\lambda(t)$. we assume a log link for an Poisson or negative Binominal distribution.

We model $\lambda(t)$ as a linear weighted combination of basis functions $B(t)$ dependent on just time.

$$
\begin{aligned}
\lambda(t) &= \sum_i w_i B(t) \\
&= \sum_i w_{i_{trend}} B_{i_{trend}} + \sum_i w_{i_{period}} B_{i_{period}}(t)
\end{aligned}
\tag{1}
$$

- 4th order Polynomial for trend, and 3rd order Polynomial for periodicity
- The design matrix $DM_{trend,period}$ for the trend and periodicity is the same for each Landkreis and concatenated for the regression. **(Dimension: [Landkreise*Days] X [order Polynomial trend + order Polynomial periodic])**
- To model the LK specific modulation we add one covraiate that contains the log of the population for each Landkreis.. This results in a Design matric DM_{LK} **(Dimension: [Landkreise*Days] X [order Polynomial trend + order Polynomial periodic + log Pop])**
- The regression we do on the model that is the composed of the two Design matrices $DM_{trend,period}$ and DM_{LK}. see Figure 20 **(Dimension: [Landkreise*Days] X [order Polynomial trend + order Polynomial periodic+ Landkreise])**
- The exp(beta) for the part of the Design matric corresponding to DM_{LK} can be compared with the population per country.

5.1.1 Trend and Periodicity - Code Matlab

In a first step the Basisfucntion for the temporal kernel is constructed.

```
    Time_vec            = 1:length(CumSumCases_Germany);
    BasisMAT_Trend      = zeros(Max_order_poly+1,length(CumSumCases_Germany));
    for Order=0:Max_order_poly
        BasisMAT_Trend(Order+1,:)=Time_vec.^(Order);
        BasisMAT_Trend(Order+1,:)=BasisMAT_Trend(Order+1,:)./(max(BasisMAT_Trend(Order+1,:)));
    end
```

In a second step the Basis functions for the periodicity for the temporal kernel is constructed. In addition corresponding basis function that are constant (i.e. BasisMATPeriodicAv) with the average value of the original basis with a periodicity are computed. These will be used later, to replace the periodic kernel with the constant kernel to model the temporal changes cleaned from the periodic component. The basis for the periodic kernel is DesignMatrix, while DesignMatrixNOP contain the Basis for the cleaned temporal modulation.

```
    Time_vec            = 1:size(CumSumCases_Germany,2);
    for Order=1:Max_order_Peri
        BasisMAT_Periodic(Order,:)  =(mod(Time_vec,7)+1).^(Order);
        BasisMAT_Periodic(Order,:)  =BasisMAT_Periodic(Order,:)./(max(BasisMAT_Periodic(Order,:)));
        BasisMAT_PeriodicAv(Order,:)=BasisMAT_Periodic(Order,:).*0+mean(BasisMAT_Periodic(Order,:));
    end

    Design_Matrix    = [BasisMAT_Trend ; BasisMAT_Periodic];
    Design_MatrixNOP = [BasisMAT_Trend ; BasisMAT_PeriodicAv];
    [beta,dev,stats] = glmfit(Design_Matrix' , CumSumCases_Germany,'poisson','constant','off');
    yfit    = glmval(beta,Design_Matrix','log','constant','off');
    yfitNOP = glmval(beta,Design_MatrixNOP','log','constant','off');
    Periodic_component  =  beta((size(BasisMAT_Trend,1)+1):end)'*Design_Matrix((size(BasisMAT_Trend,1)+1):end,:);
    growth_factor       =  beta'*Design_Matrix;
```

The fitting is done on the concatenated data across all landkreise. For this the next code is looping across all landkreise and concatenating the design matrices. The Design Matrix Design$_M atrixhastextbf(Dimension:[Landkreise*Days]X[orderPolynomialtrend+$

The individual Kernels for all age groups can be seen in the appendix. It needs to be dicusses whether we want age group specific temporal kernels. At the moment I would say no.

```
    target_vec      =[];
    Design_Matrix   =[];
    for IDXLK=1:size(CumSumDeath_LK,1)
        Indiv_Growth    = ones(1,size(CumSumDeath_LK,3))*LogPopulation(IDXLK,IDXAge);
        act_cases       = squeeze(CumSumCases_LK(IDXLK,IDXAge,:));
        if sum(act_cases)~=0
            DesignMat_part  = [BasisMAT_Trend ; BasisMAT_Periodic];%BasisMAT_Trend(1:end,:);
            %DesignMat_part(1,:) = DesignMat_part(1,:);
            DesignMat_part2     =[DesignMat_part ; Indiv_Growth];
            target_vec      =[ target_vec  act_cases'];
            Design_Matrix   =[ Design_Matrix  DesignMat_part2];
        else
            warning('No cases')
        end
    end
    target_vec(find(target_vec)==0)=NaN;
    target_vec(find(target_vec<0))=0;

    IDX_Trend        = (1:size(BasisMAT_Trend,1));
    IDX_period       = (1:size(BasisMAT_Periodic,1))+size(BasisMAT_Trend,1);
    IDX_scale        = (size(BasisMAT_Periodic,1))+size(BasisMAT_Trend,1)+1;

    [beta,dev,stats] = glmfit(Design_Matrix' , target_vec,'poisson','constant','off');
    yfit = glmval(beta,Design_Matrix','log','constant','off');
```

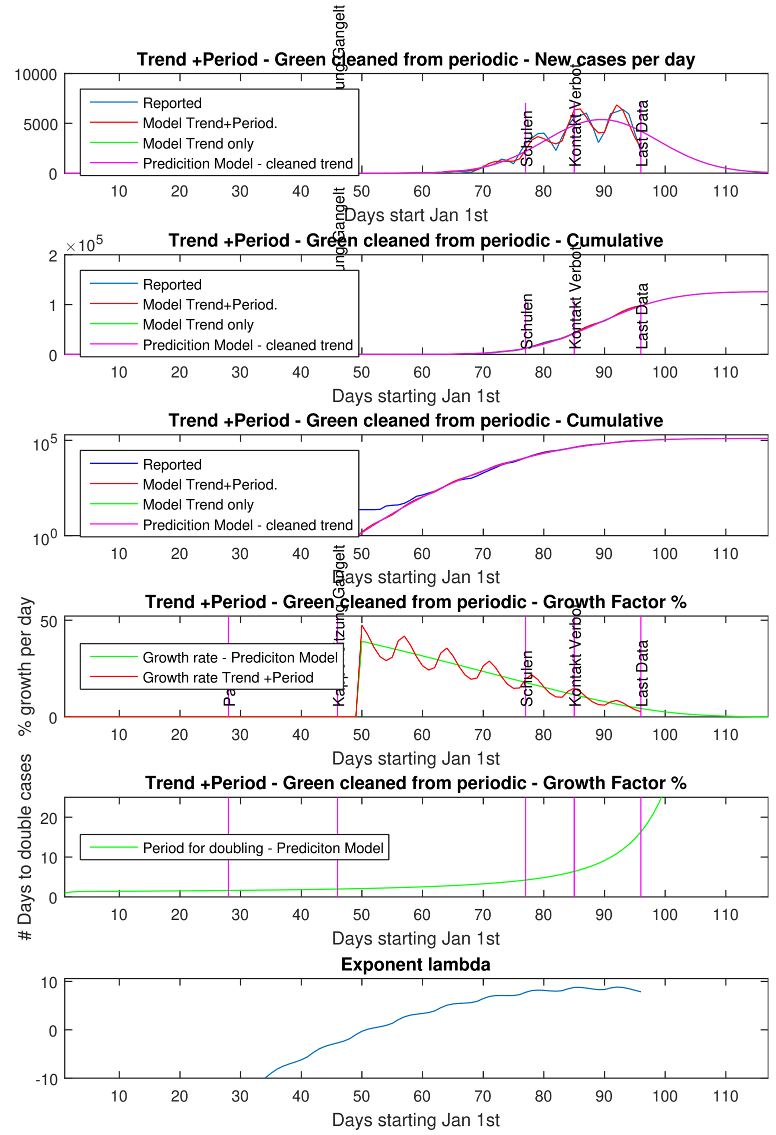


Figure 1: Polynomial model for $HW1$

```
    growth_factor     =  beta(IDX_Trend )'*BasisMAT_Trend(IDX_Trend
    ,:);
    periodic_factor   =  beta(IDX_period)'*BasisMAT_Periodic(:,:);
    periodic_cleaned  =  beta(IDX_period)'*BasisMAT_PeriodicAv(:,:);
    Offset            =  beta(end);
```

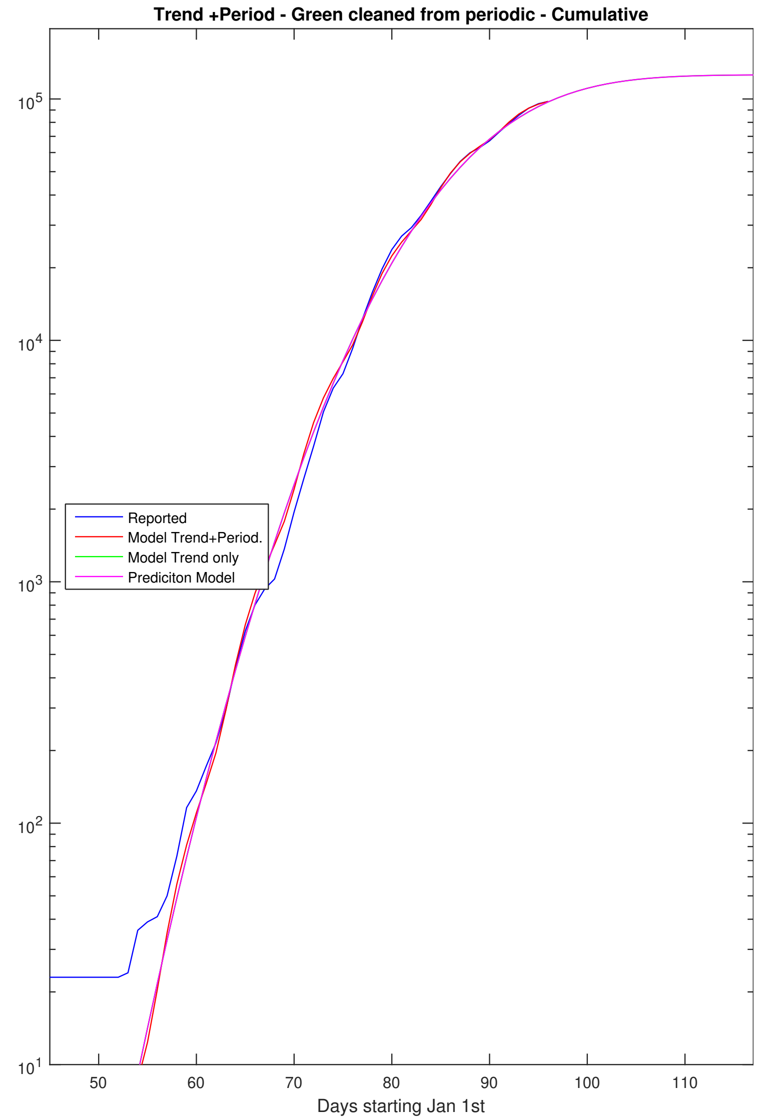


Figure 2: Logarithmic plot for $HW1$

5.2 Scaling to Generate $WH2$, $WH3$

After Model $WH1$ was modelled in a second step the temporal kernel from this model will be used and adapted.

```
            Model3                    = growth_factor+periodic_factor
     +Offset*LogPopulation(IDXLK,IDXAge);
            Model3_cleaned            = growth_factor+
    periodic_cleaned  +Offset*LogPopulation(IDXLK,IDXAge);;
```

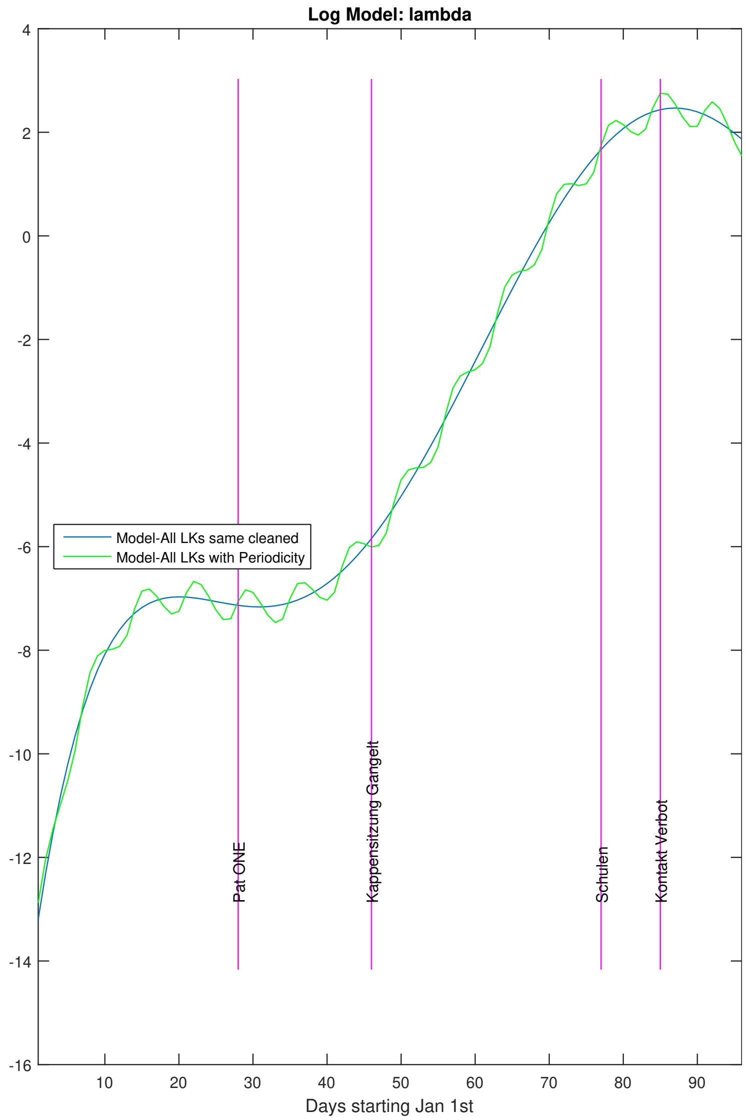


Figure 3: Temporal Kernel of model $WH1$ for age group 3. The kernel is based on a Polynomial model for $\lambda(t)$.

```
            Model3_notcleaned        = growth_factor+periodic_factor
     +Offset*LogPopulation(IDXLK,IDXAge);
            DesignMat_part(1,:)      = growth_factor.*0+1;

            target_vec2          =[ act_cases’];
            target_vec2(find(target_vec2)==0)=NaN;
            target_vec2(find(target_vec2<0))=0;
```

```
        [beta2,dev,stats]   = glmfit(DesignMat_part' , target_vec2,'poisson','constant','off','offset',Model3);
        beta_scale(IDXLK)   = beta2;
        yfit2               = glmval(beta2,DesignMat_part','log','constant','off','offset',Model3');
        yfitLK              = [yfitLK  ; yfit2];
        target_vec2_cumsum  =[target_vec2_cumsum  cumsum(target_vec2)];
        yfitLK_cumsum       =[yfitLK_cumsum    cumsum(yfit2)'];

        DesignMat_part2     = DesignMat_part;
        DesignMat_part2(2,:)= Model3;
        DesignMat_part2(3,:)= Model3.^2;
        [beta2A,dev,stats]   = glmfit(DesignMat_part2' , target_vec2,'poisson','constant','off');
        beta_scale2(IDXLK,:) = beta2A;
        yfit2A              = glmval(beta2A,DesignMat_part2','log','constant','off');
        yfitLKA             = [yfitLKA  ; yfit2A];
        yfitLKA_cumsum       =[yfitLKA_cumsum    cumsum(yfit2A)'];
```

5.2.1 Results

By Comparing the non scaled, and scaled Models we find that in a few landkreise there is a miss scaling. This can be seen based on the larger $E2$ (later deciance). In 16 one can see in row two that there are a few landkreise that are bad compared to the others if $WH1$ is used. Compared to $Wh1$ the scaled $Wh2$ in row 3 has improved results for these previously more badly modelled ones. In row 4 the improved is concnerned with an additional change in the shape of the temporal growth kernel. As a conclusion of these three plots. For the majority of the Landkreise a singel kernel can be used. For a few landkreise depicted in the figure an adaption of the shape in the early phase of the breakout is neccessary. For landkreise with early interventions this leads to changes in the later phase of the temporal kernel as well.

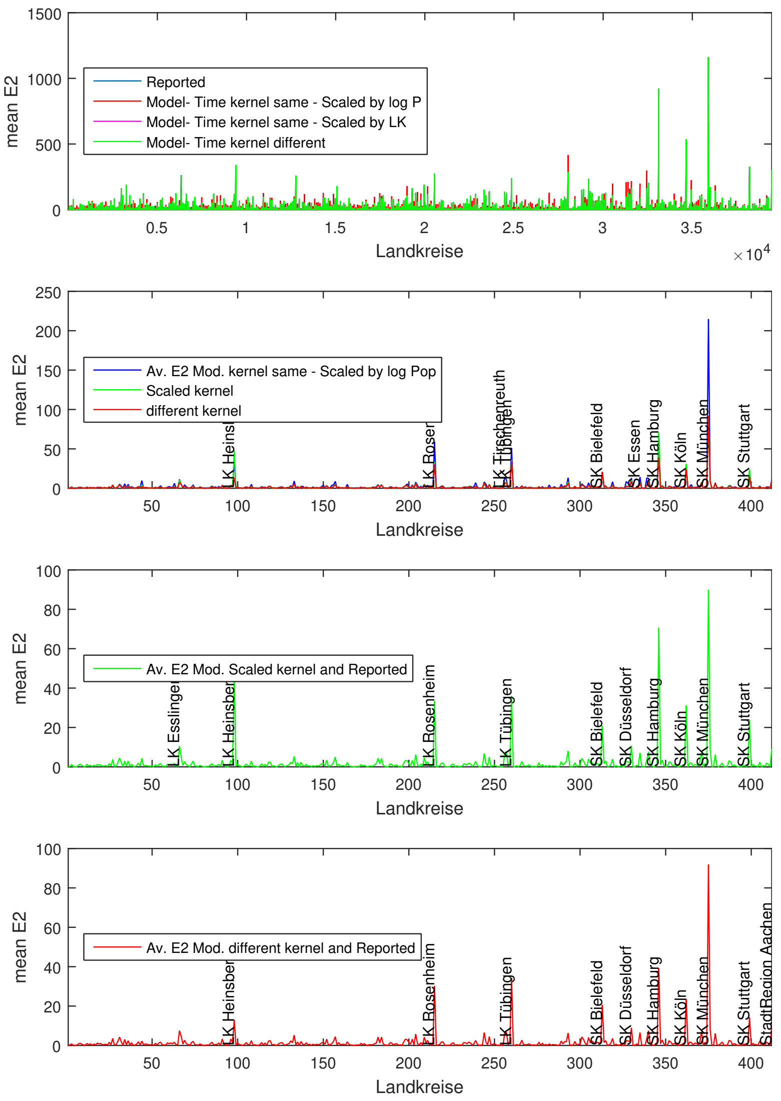


Figure 4: Temporal Kernel of model $WH1$ for age group 3. The kernel is based on a Polynomial model for $\lambda(t)$.

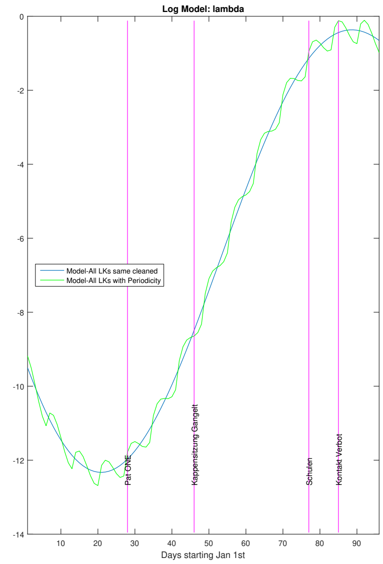


Figure 5: Temporal Kernel of model $WH1$ for age group 1. The kernel is based on a Polynomial model for $\lambda(t)$.

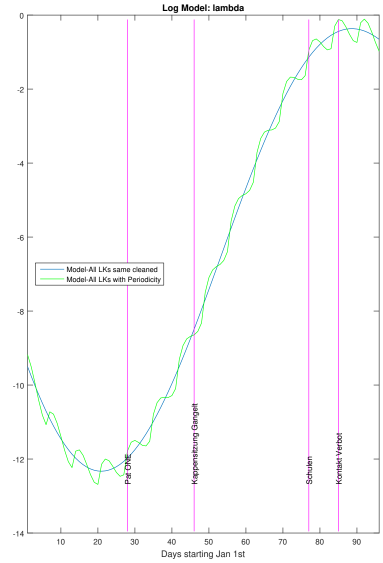


Figure 6: Temporal Kernel of model $WH1$ for age group 2. The kernel is based on a Polynomial model for $\lambda(t)$.

6 Appendix

6.1 Results

6.1.1 Polynomial time dependence of λ

The trend can be modelled by a polynomial. The advantage is that the number of extrema, infection points can be controlled by the order and that the function

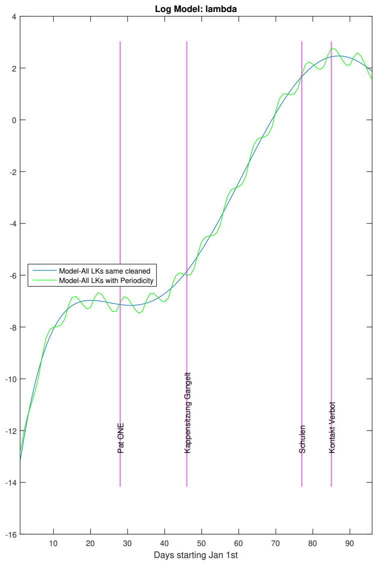


Figure 7: Temporal Kernel of model $WH1$ for age group 3. The kernel is based on a Polynomial model for $\lambda(t)$.

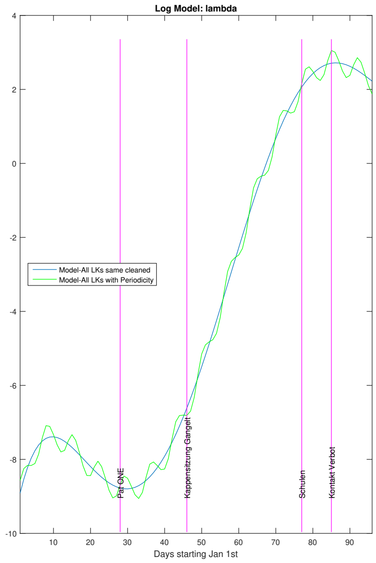


Figure 8: Temporal Kernel of model $WH1$ for age group 4. The kernel is based on a Polynomial model for $\lambda(t)$.

is global, giving the chance for an prediction with an horizon of a few days to a

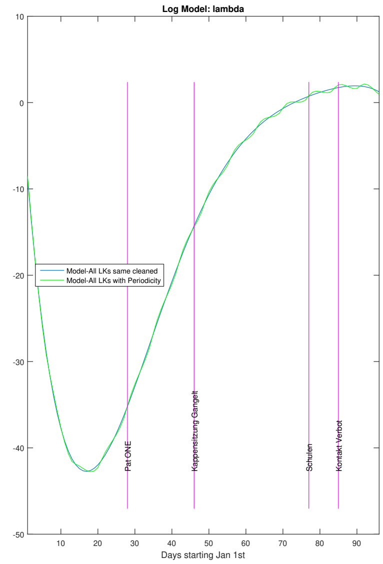


Figure 9: Temporal Kernel of model $WH1$ for age group 5. The kernel is based on a Polynomial model for $\lambda(t)$.

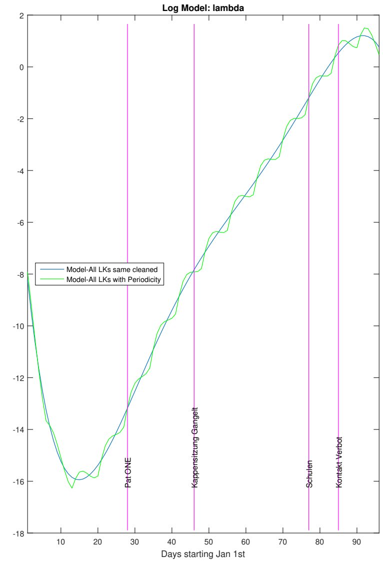


Figure 10: Temporal Kernel of model $WH1$ for age group 6. The kernel is based on a Polynomial model for $\lambda(t)$.

week.

$$w_{i_{trend}} B_{i_{trend}} = w_{i_{trend}} \cdot \left(\frac{t}{t_{max}}\right)^{i} \text{ with t between 0 and max days} \tag{2}$$

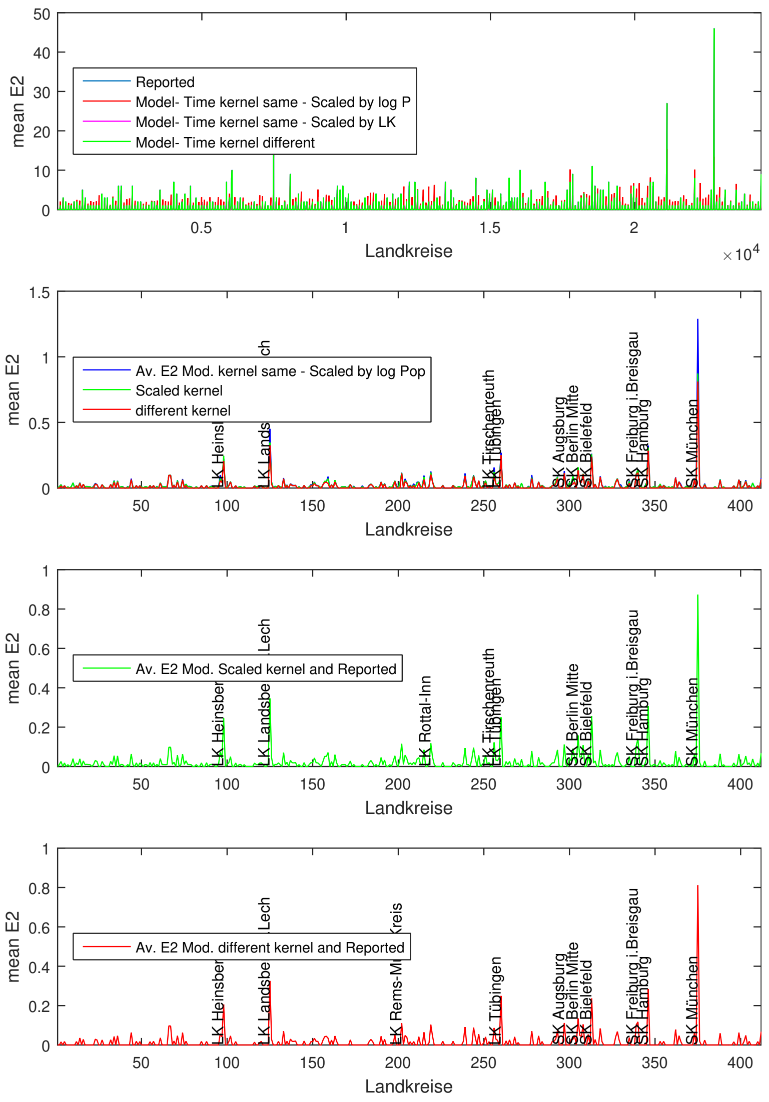


Figure 11: Temporal Kernel of model $WH1$ for age group 1. The kernel is based on a Polynomial model for $\lambda(t)$.

OBSERVATION: The performance of a polynomial is good. However due to the dip at the end, the model either requires order 3 at least, or it starts diverging much before the end observed interval, leading to an under estimation of the real number of cases.

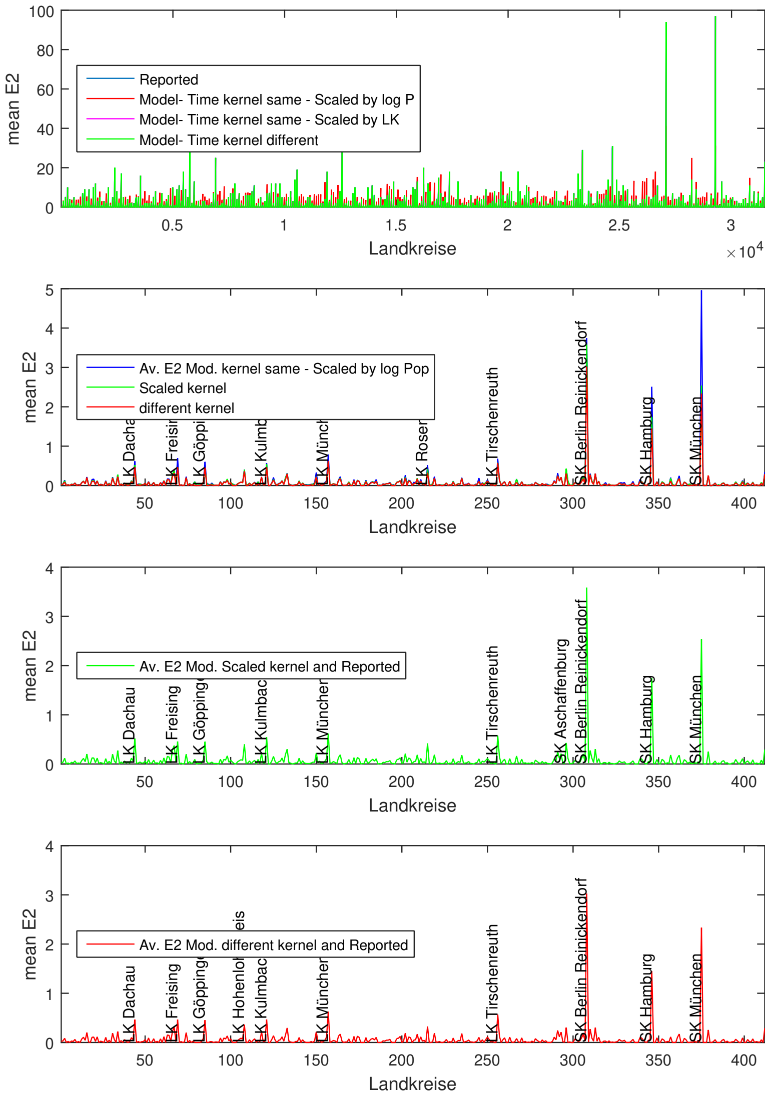


Figure 12: Temporal Kernel of model $WH1$ for age group 2. The kernel is based on a Polynomial model for $\lambda(t)$.

IDEA TO ADDRESS THIS: Either rise the exponent of the last basis function and zero most of the last component. Or add two cubic splines with a restricted support for the very last points to allow modelling this last part with strictly local basis functions.

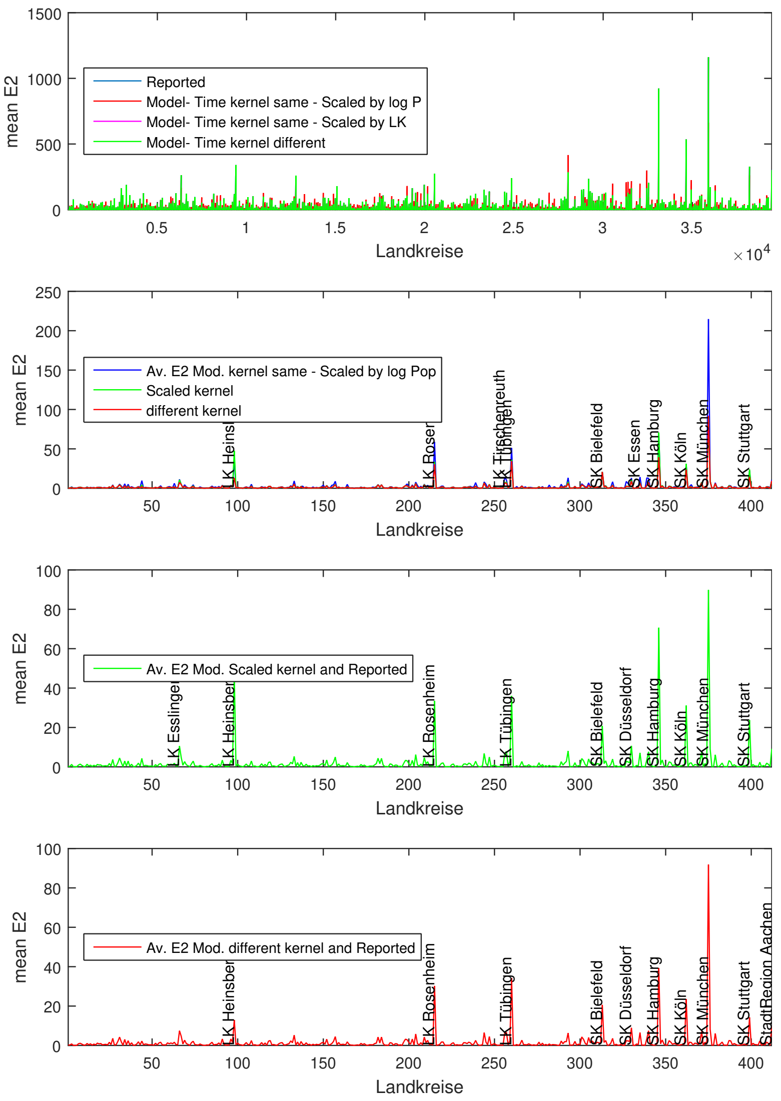


Figure 13: Temporal Kernel of model $WH1$ for age group 3. The kernel is based on a Polynomial model for $\lambda(t)$.

6.1.2 Cubic-Spline time dependence of λ

The trend can be modelled by a cubic spline basis. The advantage is that the control points can be spaced wider for the initial part. Dense knots can be at the end, to model the dip. the number of knots controls the number of changes in the growth rate. The limited support of the local basis function allows for generating alternative models, to answer questions like. How would the situation

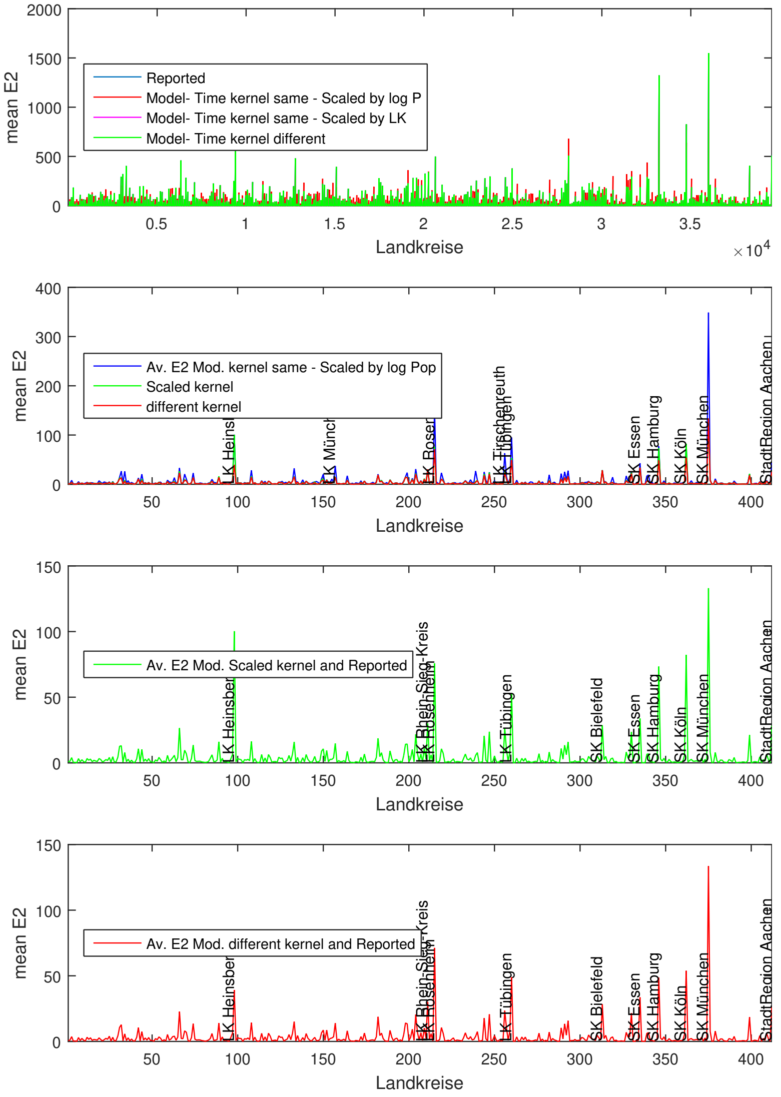


Figure 14: Temporal Kernel of model $WH1$ for age group 4. The kernel is based on a Polynomial model for $\lambda(t)$.

look today if the growth would have been like in the previous week.

$$w_{i_{trend}} B_{i_{trend}} \text{with first knot at 45 days} \tag{3}$$

The knot spacing can be chosen wide at the beginning. First knot at day 45 was a good choice. Than a equals spacing of the order 14 days gives good results.

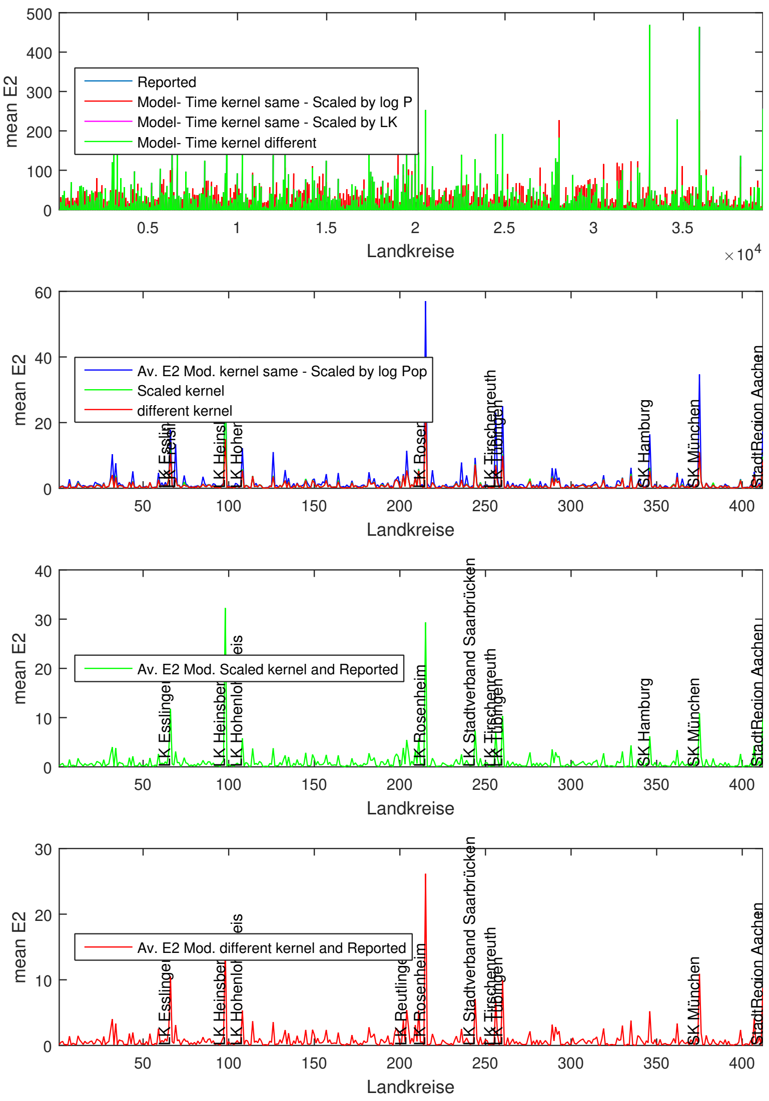


Figure 15: Temporal Kernel of model $WH1$ for age group 5. The kernel is based on a Polynomial model for $\lambda(t)$.

This knot spacing needs to be larger than the weekly periodicity to separate time scales. To model the dip at the end two additional knots at end-4 and end-2 gave good fits, considering that the dip is on the order of 4 days.

OBSERVATION: The locality of the model prevents prediction into the feature.

IDEA TO ADDRESS THIS: ?)-; Do not look into the future.

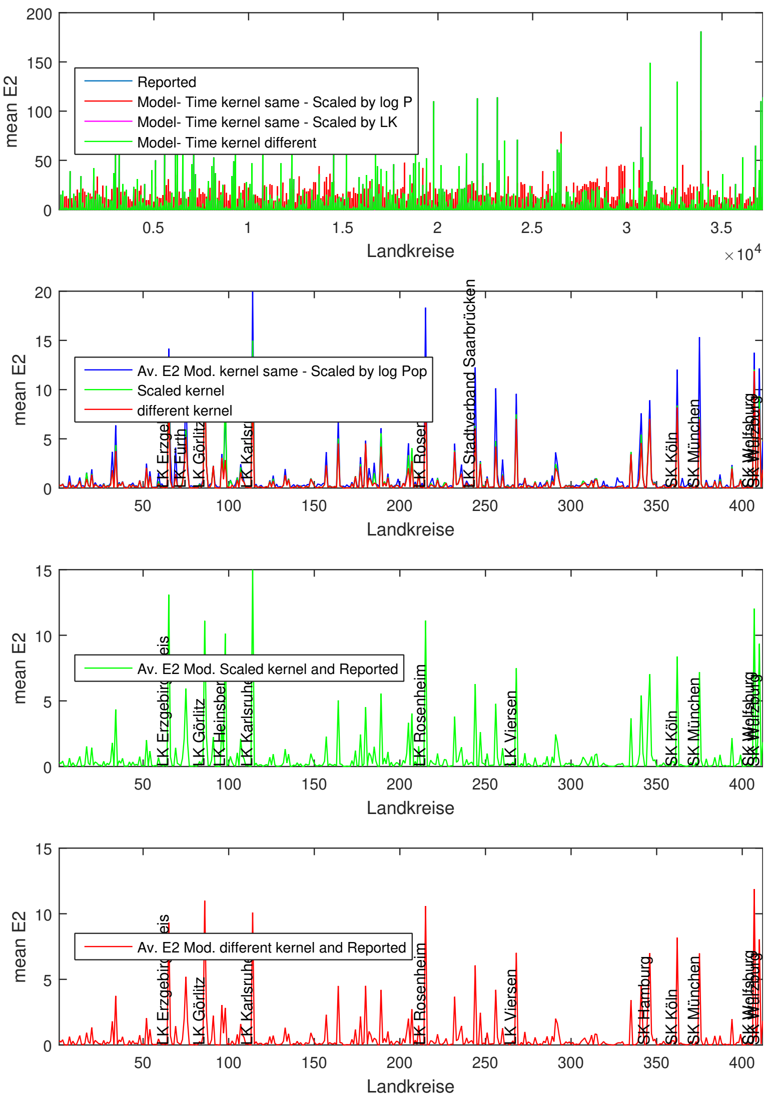


Figure 16: Temporal Kernel of model $WH1$ for age group 6. The kernel is based on a Polynomial model for $\lambda(t)$.

6.1.3 Polynomial periodicity λ

The periodicity can be modelled by a polynomial. It is simple and works well. Polynomial order is 3.

$$w_{i_{period}} B_{i_{period}} = w_{i_{period}} \cdot \left(\frac{t}{7}\right)^{i} \quad \text{t is days modulo 7} \tag{4}$$

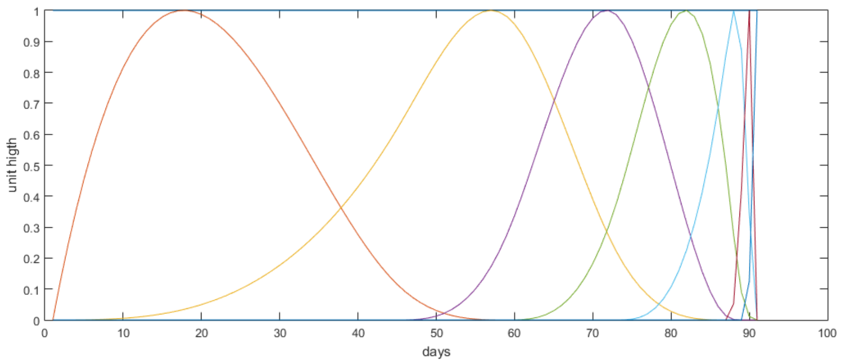


Figure 17: Spline Basis function to model the time dependence of the exponent $\lambda(t)$

OBSERVATION: The performance of a polynomial is good. However due to the dip at the end, the model either requires order 3 at least, or it starts diverging much before the end observed interval, leading to an under estimation of the real number of cases.
IDEA TO ADDRESS THIS: Either rise the exponent of the last basis function and zero most of the last component. Or add two cubic splines with a restricted support for the very last points to allow modelling this last part with strictly local basis functions.

6.2 Modelling Comparisons

6.2.1 Poly nominal versus Cubic Splines

Both the polynomial and the Cubic Basis splines give very similar results. The cubic splines require a few more parameters to span the entire range. Concerning the overall quality of the fit, both methods are comparable. Given that the implementation based polynomial appears simple in respect to Python we will model the data with a polynomial.

7 Additional Hypothesis Testing And Model Building

Based on the Model Goals we decided on the following Model sequence and Model Comparisons.

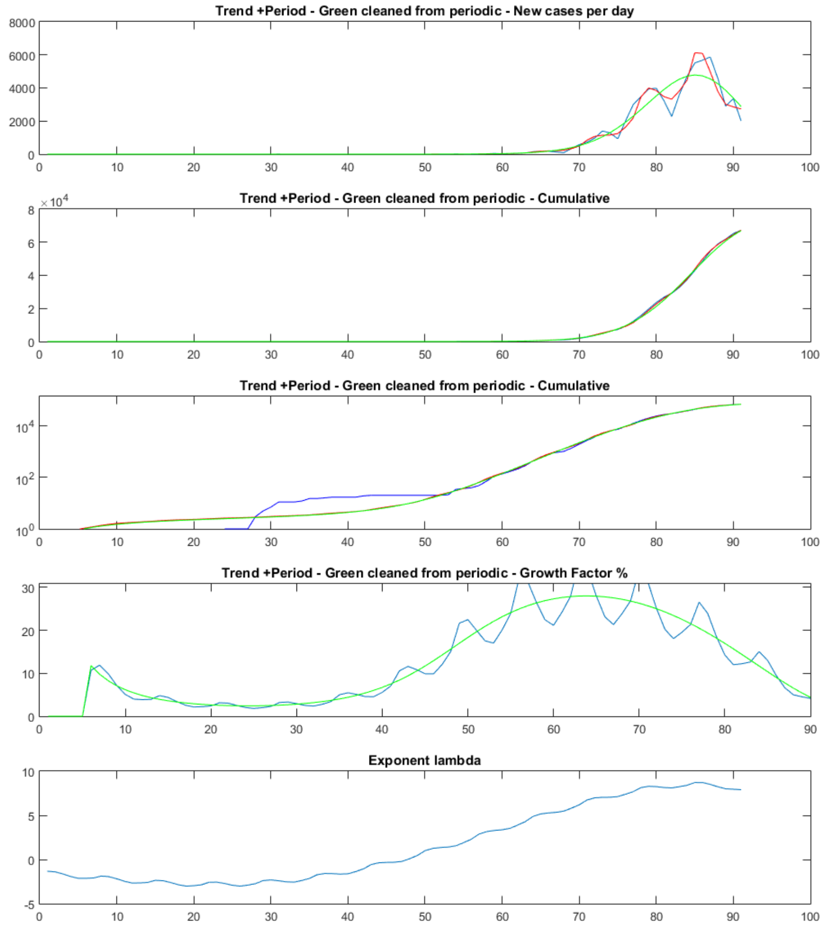


Figure 18: Fit based on polynomial of order 4 for $\lambda(t)$. **ROW1:** Daily new cases. (blue) real reported numbers (red) model including trend and periodic component (green) model cleaned from periodic part. **ROW2:** same as row 1 but cumulative cases are plotted **ROW3:** same as row2 on log-scale. **ROW4:** Computed growth rate in percent between days. This estimated is estimated by the local slope of the log of the cumulative counts. (blue: with periodic component, green:cleaned from periodic component) **ROW4:** Exponent $\lambda(t)$

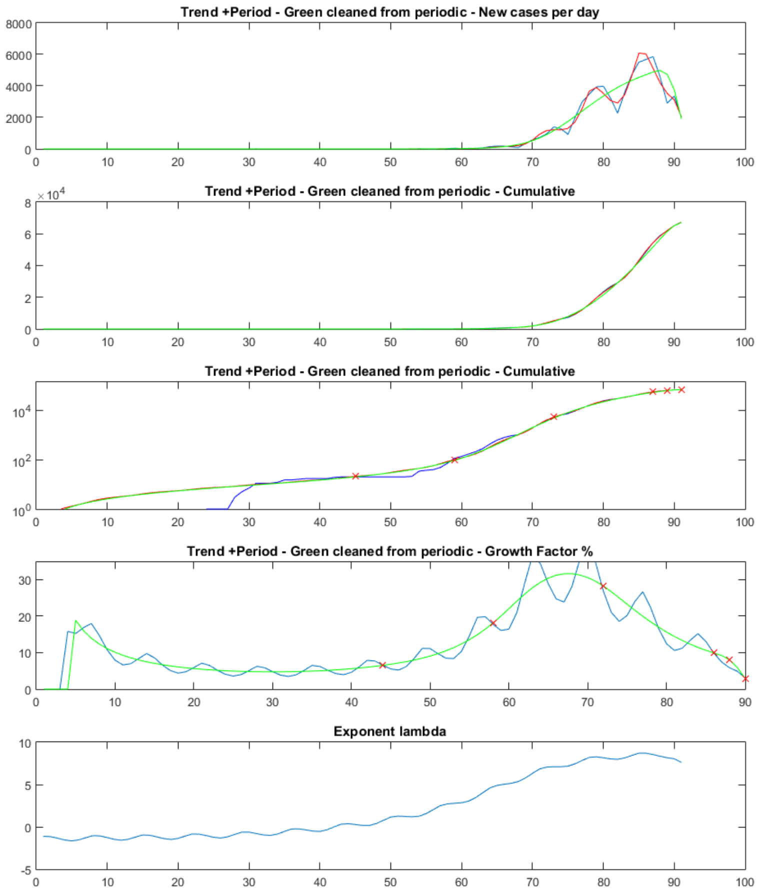


Figure 19: Fit based on Splines placed at days [1,45,59,73,87,89,91] for $\lambda(t)$. **ROW1:** Daily new cases. (blue) real reported numbers (red) model including trend and periodic component (green) model cleaned from periodic part. **ROW2:** same as row 1 but cumulative cases are plotted **ROW3:** same as row2 on log-scale. **ROW4:** Computed growth rate in percent between days. This estimated is estimated by the local slope of the log of the cumulative counts. (blue: with periodic component, green:cleaned from periodic component) **ROW4:** Exponent $\lambda(t)$

7.1 Hypothesis 1 and Alternative Hypothesis 1

Hypothesis 1 assumes that the dynamics of the COVID-19 growth is identical for all Landkreise. However, we assume that the speed of growth is different.

This can account for different population densities for example. It is important to note that this model, will does not require population density as an input into the model, but the landkreis specific parameter can be compared to the population density.

The **alternative Hypothesis 1** assumes that the principle dynamics is not the same. We will test this by modelling the corresponding Model for the Hypothesis 1 at first, and derive the alternative Model from this in a second step. The alternative allows for a modification of the principle growth shape.

- By comparing the two fits of the original and the alternative model we can assess in how many cases the assumption that the growth is identical across all landkreise is the justified.
- By comparing the two fits of the original and the alternative model we can identify these LK that strongly deviate from the average growth. These differences can be compared to policy changes.
- By comparing in a scatter plot the parameters of the original model 1 against the populations density of the respective LK we can assess the relevance of population density for explaining the data.

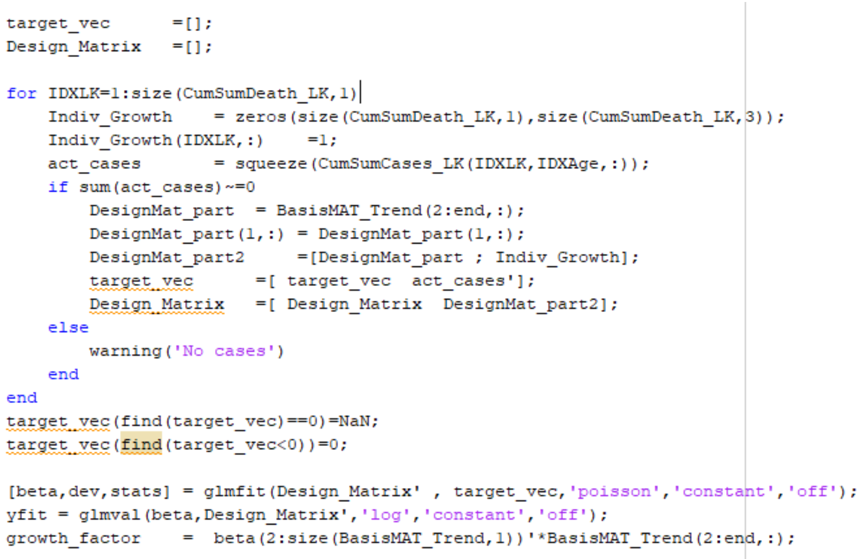
Observation Hypothesis 1: The large number of parameters can be handeled by a ML approach and gordon modelled it successfully. However the framework defined by the source code from Johannes and Olivera, where the trend is not LK specific we should not use model 1 but instead use model Hypothesis 3.

7.2 Hypothesis 2 and Alternative Hypothesis 2

Hypothesis 2 same as **Hypothesis 1**

The **alternative Hypothesis 2** assumes that the interaction affects between LK explains part of the residuals (compared to the observed data) of the model corresponding to Hypothesis 2.

- By comparing the two fits of the original and the alternative model we can assess whether and to what degree using the interaction kernel improved the model.
- By the shape of the interaction kernel we can assess the range of interaction and the speed of the spread.
- By the sensitivity analysis we can compare the importance of the interaction in contrast to the trend to check whether travel restrictions are indicated.



```
target_vec      =[];
Design_Matrix   =[];

for IDXLK=1:size(CumSumDeath_LK,1)
    Indiv_Growth     = zeros(size(CumSumDeath_LK,1),size(CumSumDeath_LK,3));
    Indiv_Growth(IDXLK,:)     =1;
    act_cases        = squeeze(CumSumCases_LK(IDXLK,IDXAge,:));
    if sum(act_cases)~=0
        DesignMat_part   = BasisMAT_Trend(2:end,:);
        DesignMat_part(1,:) = DesignMat_part(1,:);
        DesignMat_part2     =[DesignMat_part ; Indiv_Growth];
        target_vec      =[ target_vec  act_cases'];
        Design_Matrix   =[ Design_Matrix  DesignMat_part2];
    else
        warning('No cases')
    end
end
target_vec(find(target_vec)==0)=NaN;
target_vec(find(target_vec<0))=0;

[beta,dev,stats] = glmfit(Design_Matrix' , target_vec,'poisson','constant','off');
yfit = glmval(beta,Design_Matrix','log','constant','off');
growth_factor    =  beta(2:size(BasisMAT_Trend,1))'*BasisMAT_Trend(2:end,:);
```

Figure 20: Matlab Code composing the Indicator matrix and joing both. The design matrix BasísMAT Trend is $DM_{trend,period}$. The design matrix Indiv Growth is DM_{LK}

8 Appendix

9 Appendix

References

[1] Olivera Stojanović, Johannes Leugering, Gordon Pipa, Stéphane Ghozzi, and Alexander Ullrich. A bayesian monte carlo approach for predicting the spread of infectious diseases. *PLOS ONE*, 14(12):1–20, 12 2019.

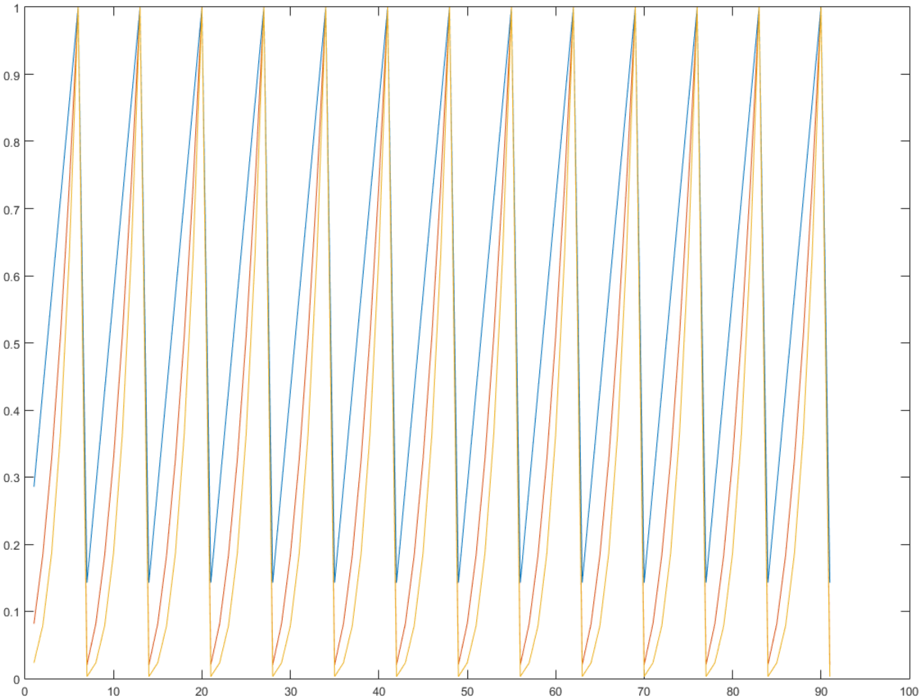


Figure 21: periodic basis