Excercise 4 - Forecasting in energy systems

Models for the heat dynamics of a building

Marco Hernandez Velasco

August 2018

In this exercise you will work with setting up models, which are useful for many applications of forecasting in energy systems. The models are setup in a two-stage approach, first transformations of the inputs are applied and then an estimation method is applied.

The exercise deals with first load forecasting, then solar and wind forecasting. It starts by introducing a simple linear low-pass filter for input transformation (and base splines) and then this is used together with linear regression for load forecasting. Then the recursive least squares (RLS) estimation method is introduced, and applied for load forecasting.

For solar and wind forecasting both base spline and kernel methods are used.

In the exercise numerical weather predictions (NWPs) are used as model input. The first step is to understand how they are set up. They are set up as matrices. It holds for each time t the latest available forecasts along the row for the variable where * t is the counter of time for equidistant time points. In this notation normalized, such that the sampling time is $t_1 - t_0 = 1$ (the time stamps are then just kept in another vector)

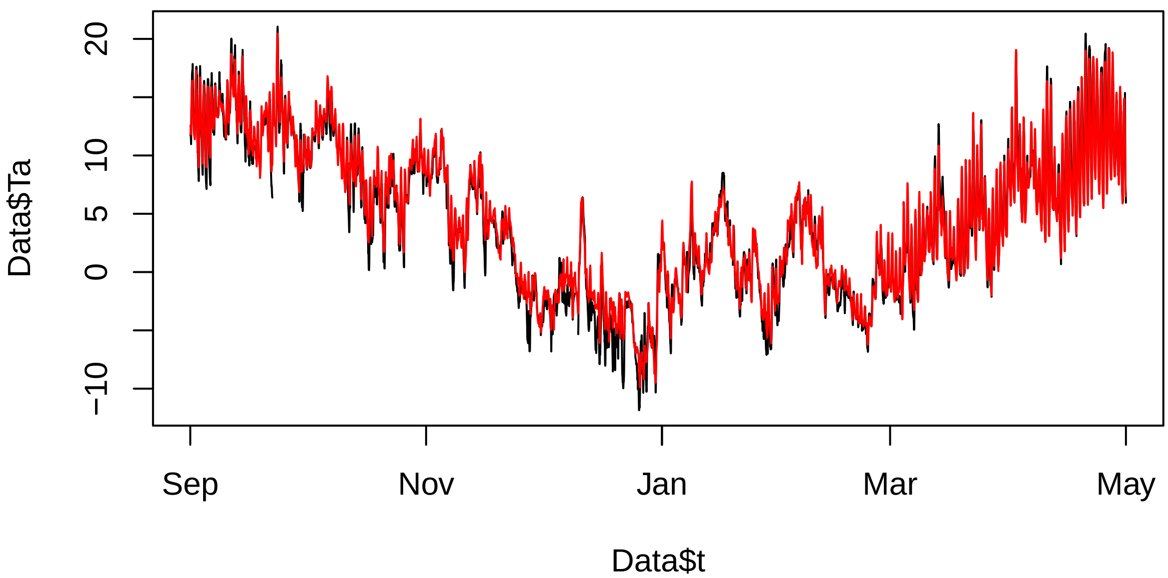
- t_0 is the first available time point
- n_k is the length of the forecasting horizon
- The column names are indicated above the matrix, they are simply a k concatenated with the value of k.

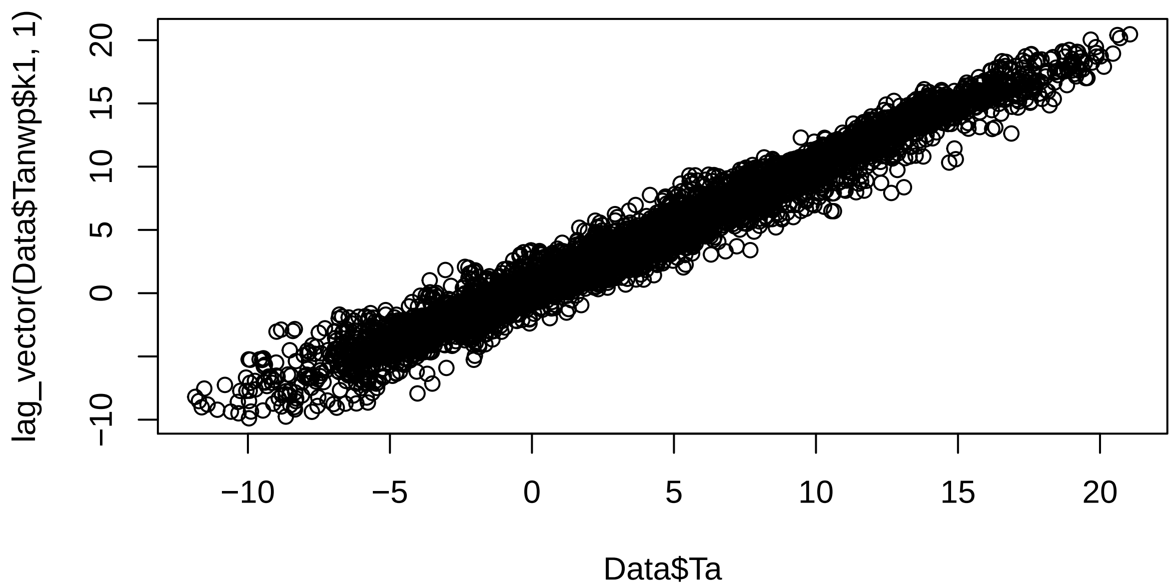
The time t can then just be thought of as an index (in the R code we use just i and the time stamps are kept in a vector).

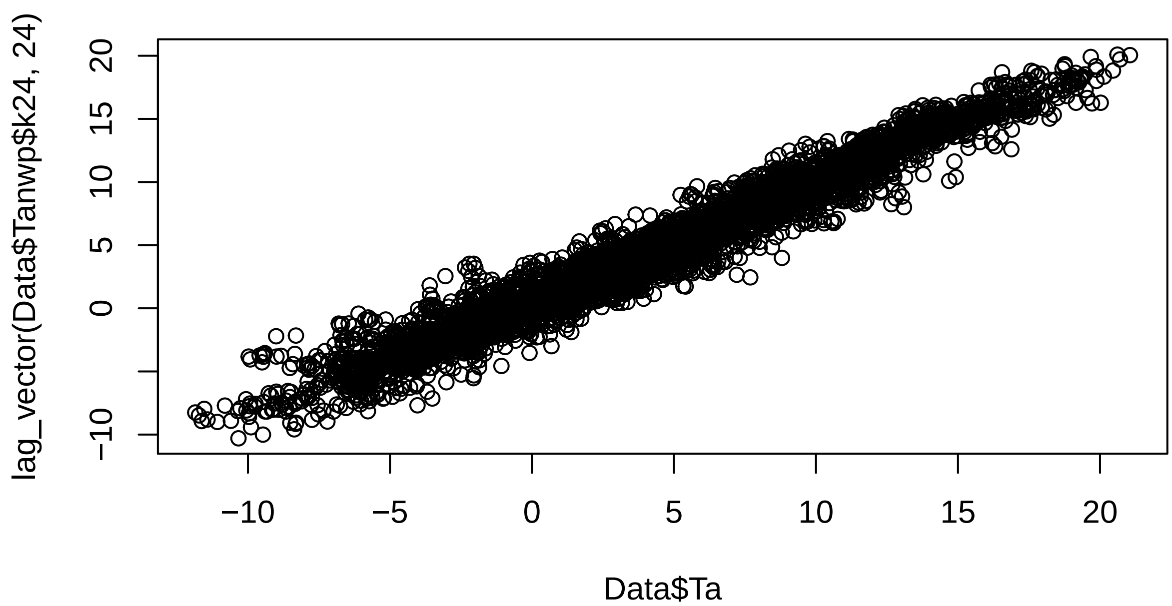
Q1 - Data setup and linear regression

In the exercise numerical weather predictions (NWPs) are used as model input. The first step is to understand how they are set up: as matrices. The main point is, that in order to fit a model for the k'th horizon you will need to lag the forecast input.

Now the point is, that if we want a forecast model for k steps ahead, then we can simply use lm() in R on this data.





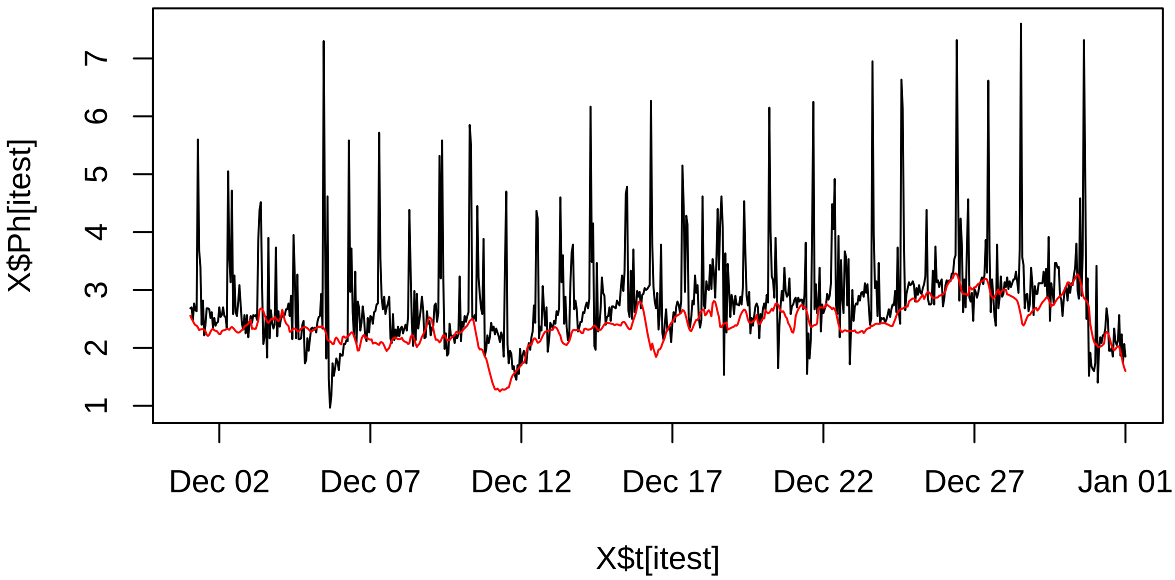


- Does the forecast look reasonable? Which seems to be most accurate k = 1 or 24 steps ahead?

Yes, the weather forecast seems reasonable even with a k = 24, it doesn't change too much and fits relatively good to a straight line, altough the best restult is reached in this case with k = 1

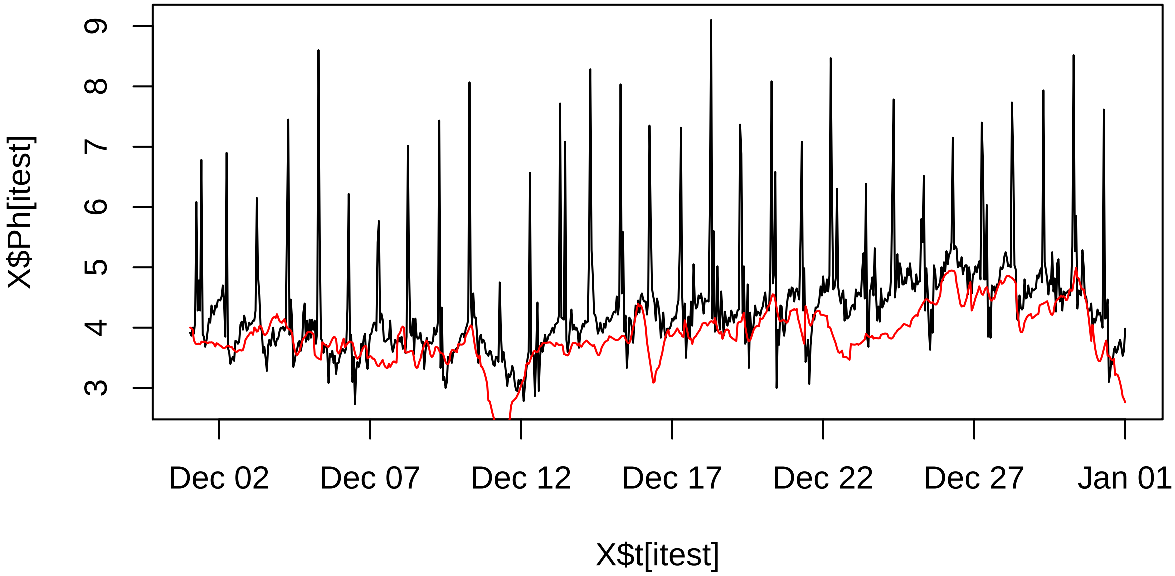
- Try to learn how the data is setup, divide into a training and a test set, fit a linear regression model for $k = 1$ step ahead. Does the forecast look reasonable?

For House 4, with k = 1 steps ahead, the load forescast is not so good and there is still alot of variation not being considered. All the coefficients are significant and the model is able to account for 49% of the variance (R^2) and has a low RMSE compared to the other houses.



- Try to calculate a forecast for House 5 with k = 36 steps ahead.Give an example and a summary of what you find.

For House 5, with k = 36 steps ahead, the load forescast is not so good and there is still alot of variation not being considered. The coefficient for the solar radiation (G) is not significant in the linear model of house 5 (perhaps there are no windows in this house?). However the model is able to account for 65.7% of the variance (R^2) and has a higher RMSE compared to house 4 with k=1.



X$Ph[itest]
3
4
5
6
7
8
9
Dec 02
Dec 07
Dec 12
Dec 17
Dec 22
Dec 27
Jan 01
X$t[itest]

Q2 - Low-pass filter

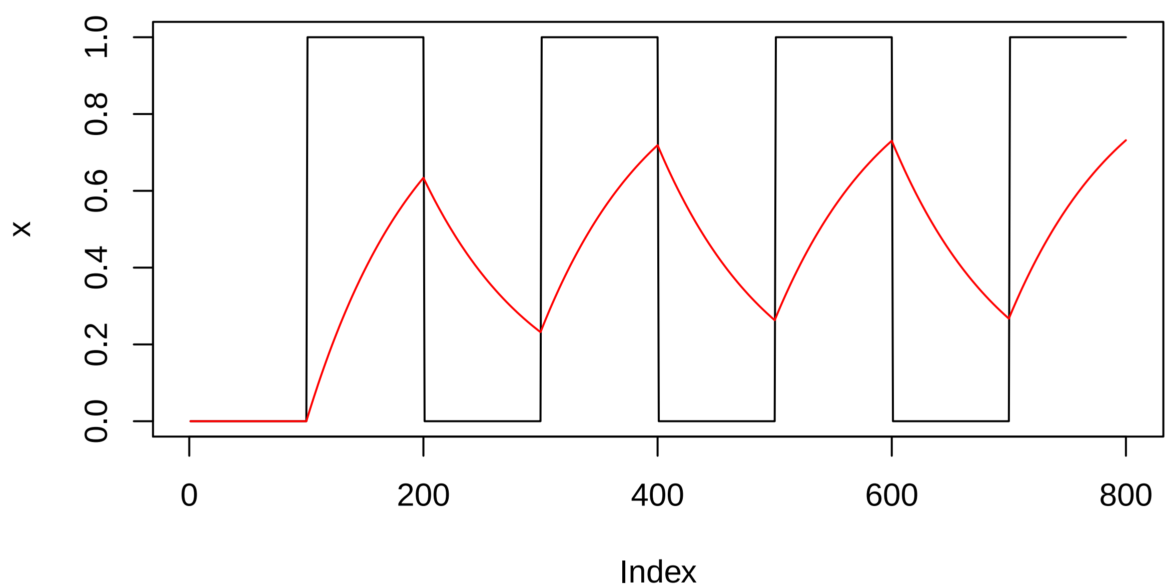
Now we do know that there are dynamics, such that the heating doesn't change immediately when the ambient temperature change, that's why we usually will use a time series model (discrete ARMAX or continuous GB), however here we introduce a slightly simplified way to do it.

We know, that the response of a building can be modelled as an R-C network, which lead to a low-pass filtering effect. Hence, we can apply a low-pass filter to the input, and the use that in the linear regression.

In order to take dynamics into account we can filter the inupts. One can say it is a transformation of the inputs (like with base splines).

First in order to model a linear dynamical 1st order system (i.e. single RC) make a sequence like an on/off signal. It is the simplest first order low-pass filter with stationary gain of one:

$$H(B) = \frac{1-a_1}{1-a_1 B}$$



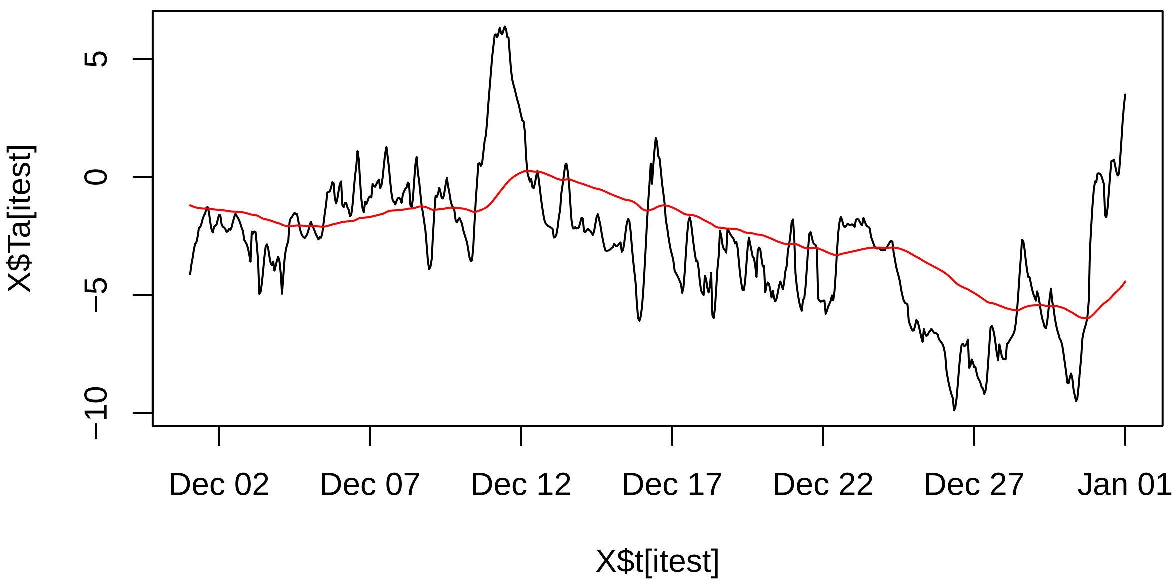
- What happens with the relation between the input and the low-pass filtered signal e.g. what is the relation between the time constant and a1?

The low-pass filter smoothness the signal and makes the sudden on/off changes slower. When a1 takes a value of 1 then the filtered signal is always 0. When $a_1 = 1$, then the filtered signal follows the input exactly. The time constant is how long it takes to reach the 96% of the input signal.

- What about the stationary gain? (i.e. the limit y approaches)

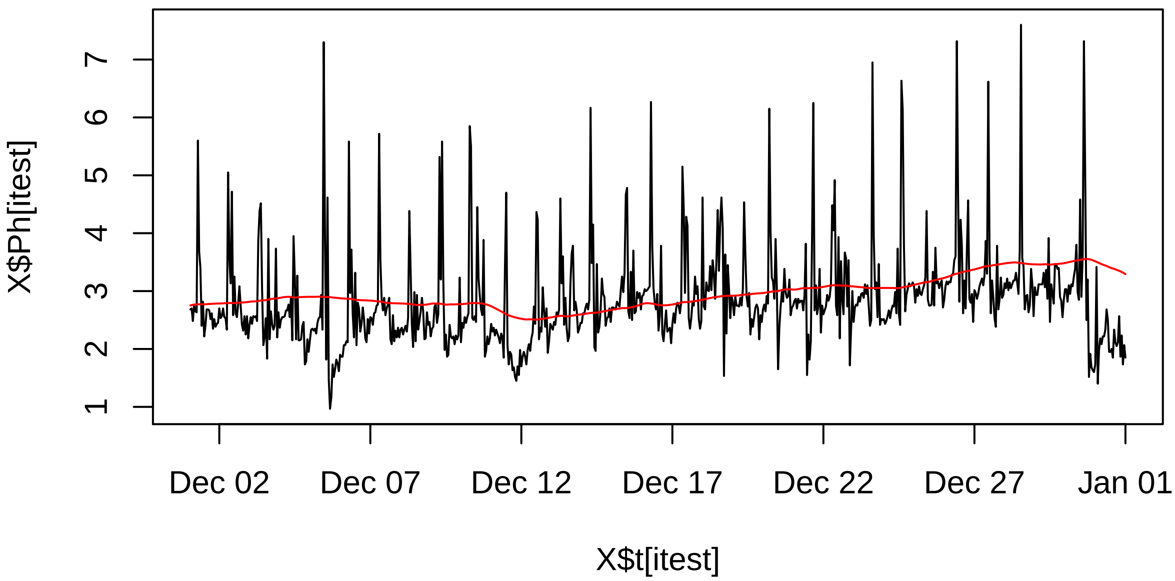
Depending on the value of a1, the filtered signal doesn't reach 0 after the cycle and there is a gain which reaches a limit.

Q3 - Load forecast



- Is the linear model tuned for the particular building heat dynamics?

It is tuned for the thermal mass of the buidling following the trend of the outdoor temperature but doesn't follow properly the fluctuations.



- Are the coefficients significant?

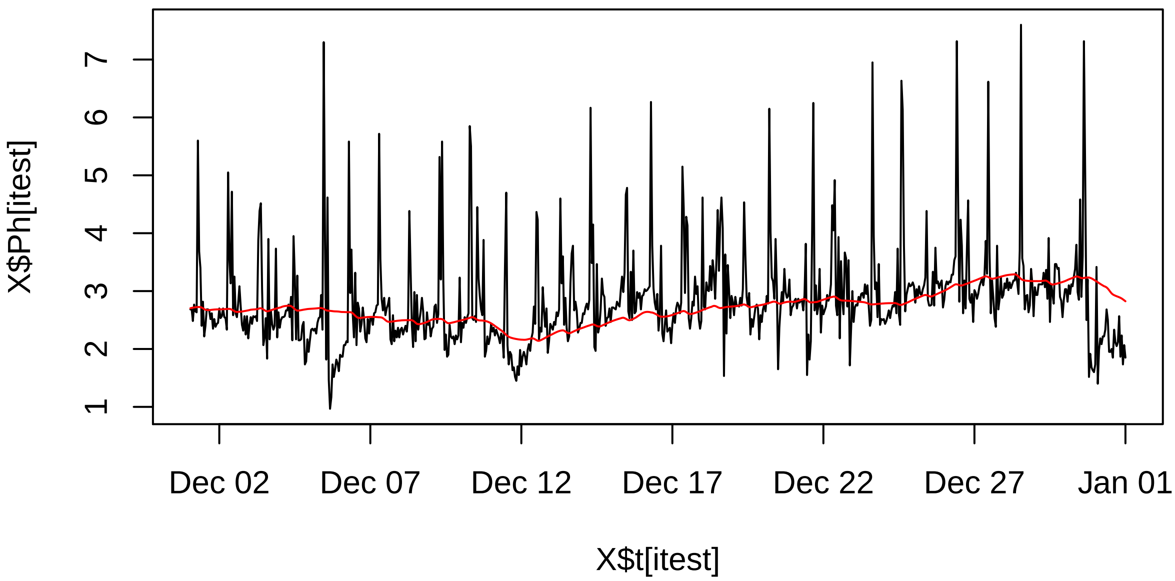
The coefficient for the ambient temperature (Ta) is significant but not the one for the solar radiation (G).

- Are the forecasts improved in terms of RMSE?

Yes, the RMSE improved from 0.932183 without low-pass filter to 0.8455964.

Tunning the low-pass coefficient:

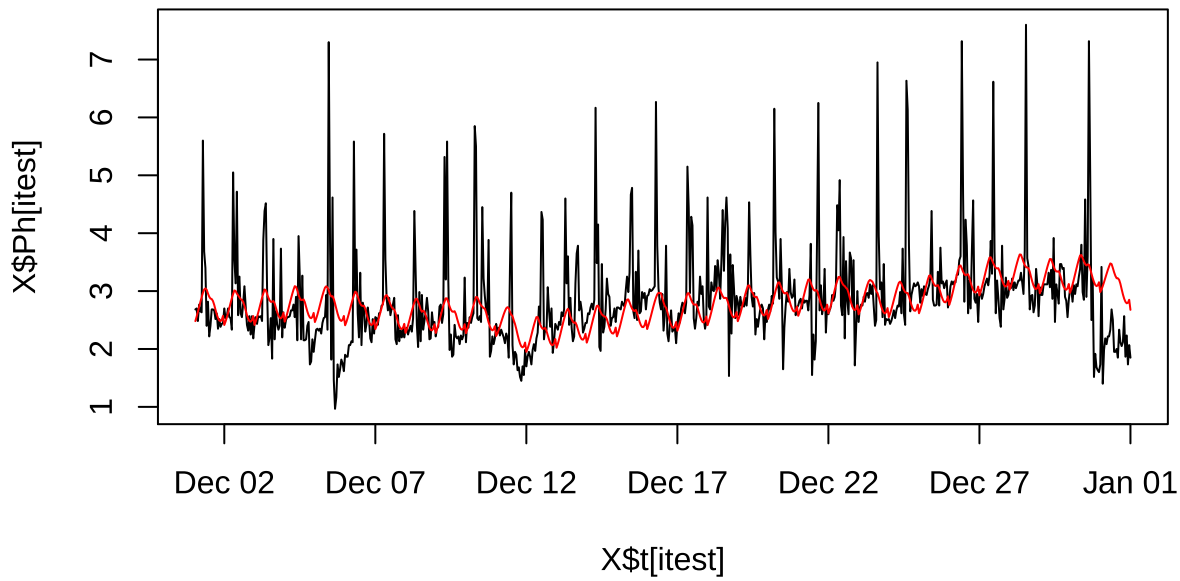
In order to tune the low-pass filter coefficients (one for the ambient temperature and one for the solar radiation) apply an optimizer to minimize the RMSE with ***Leave-one-out cross-validation*** on the test set.



- Did the model improve?

There is a minor improve in the RMSE from 0.846 to 0.834 with the tuned low-pass filter.

Finally, include a diurnal curve using base splines. Make the base splines using bs().



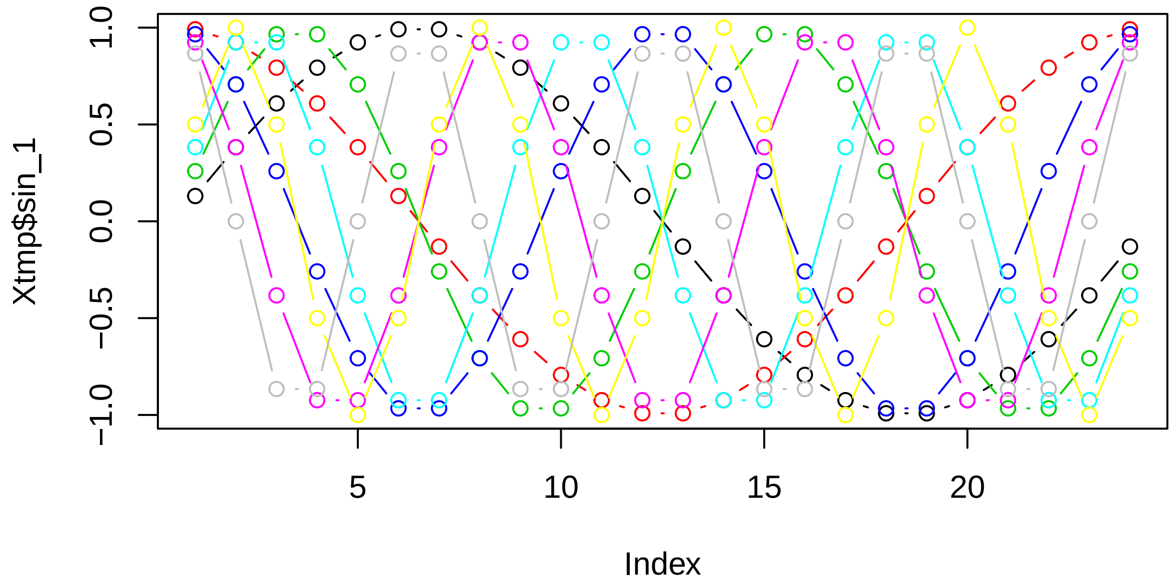
- Do we get better forecasts?

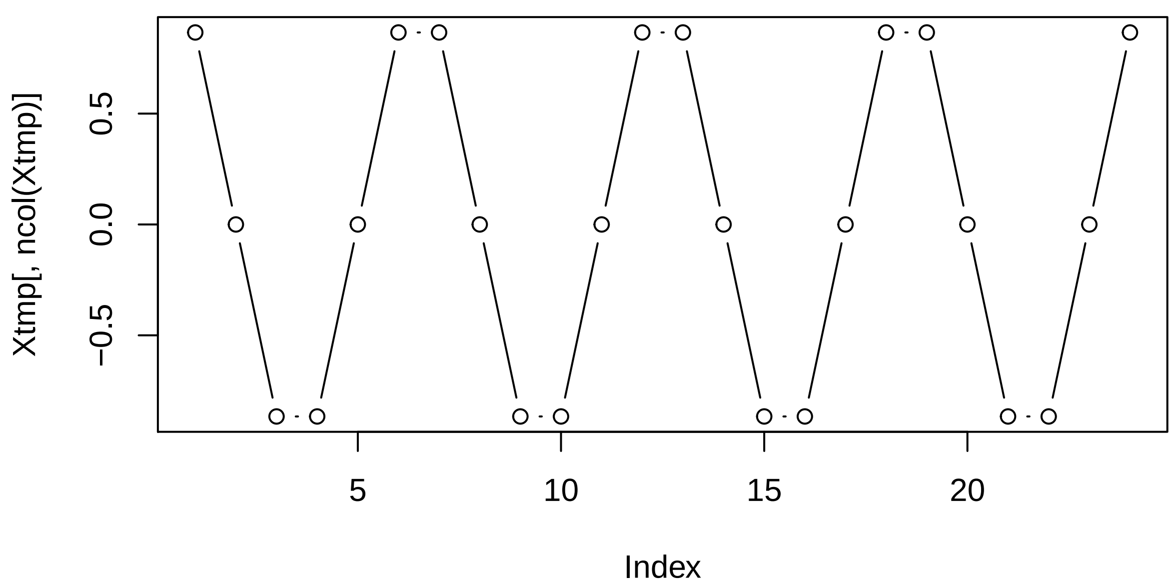
Yes, including the splines for the diurnal curve improved the RMSE reducing it to 0.789.

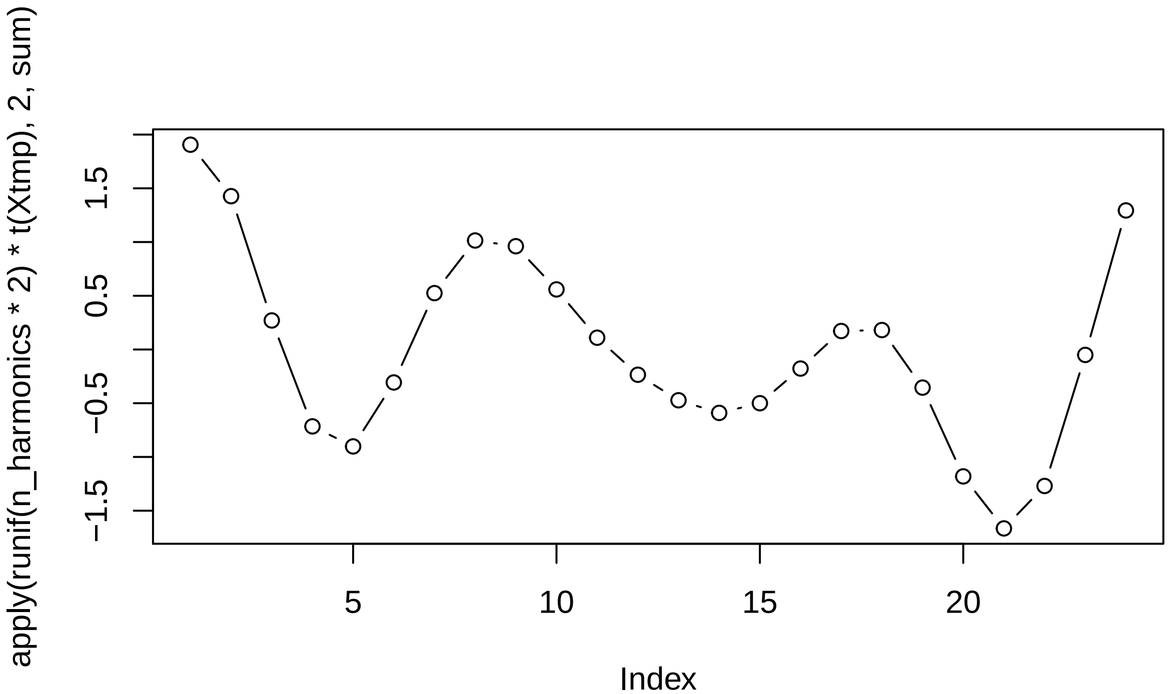
- How to choose the degrees of freedom df? maybe use AIC or BIC?

An optimization option can be made minimizing the AIC, which already give penalty to the number of parameters (see Peder's presentation for forecast optimization and Bacher and Madsen [2011].)

Use Fourier series instead as basis functions for the diurnal curve *(Optional)*.

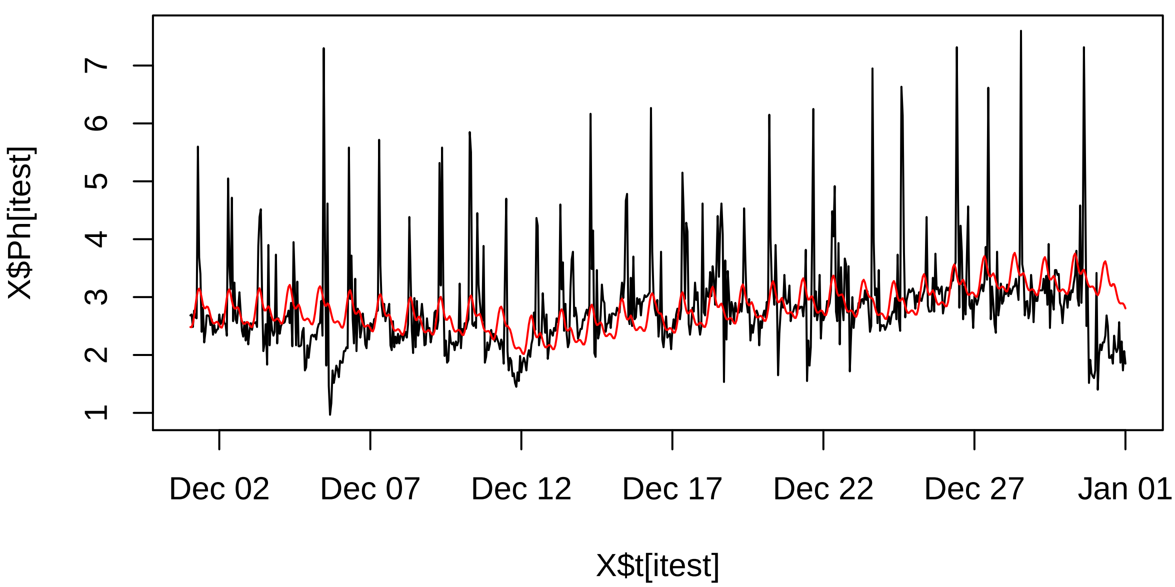

Xtmp$sin_1
−1.0
−0.5
0.0
0.5
1.0
5
10
15
20
Index


Xtmp[, ncol(Xtmp)]
−0.5
0.0
0.5
5
10
15
20
Index

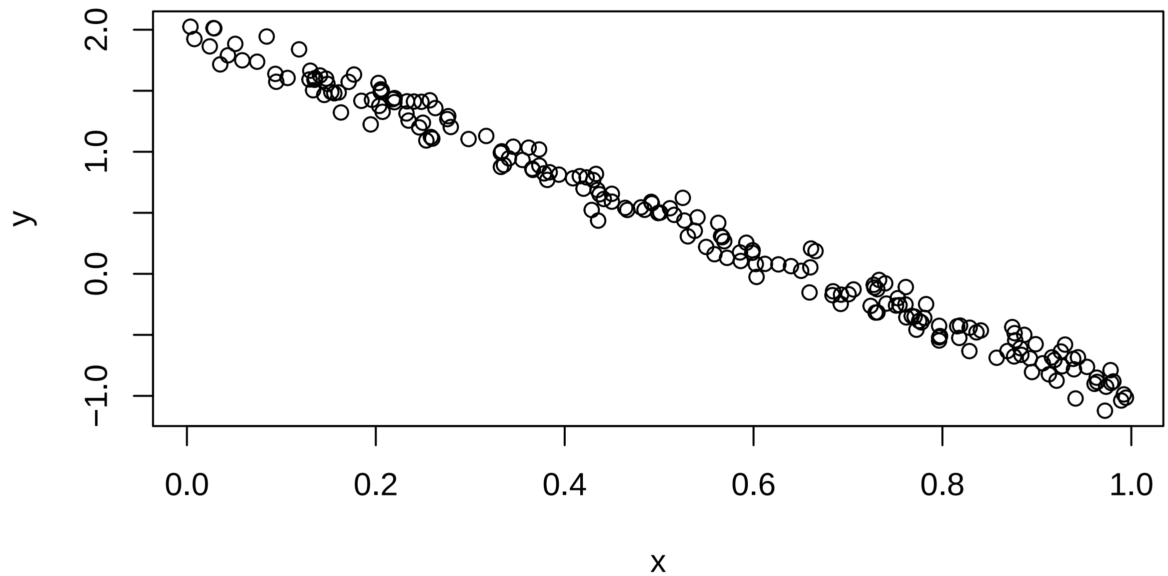


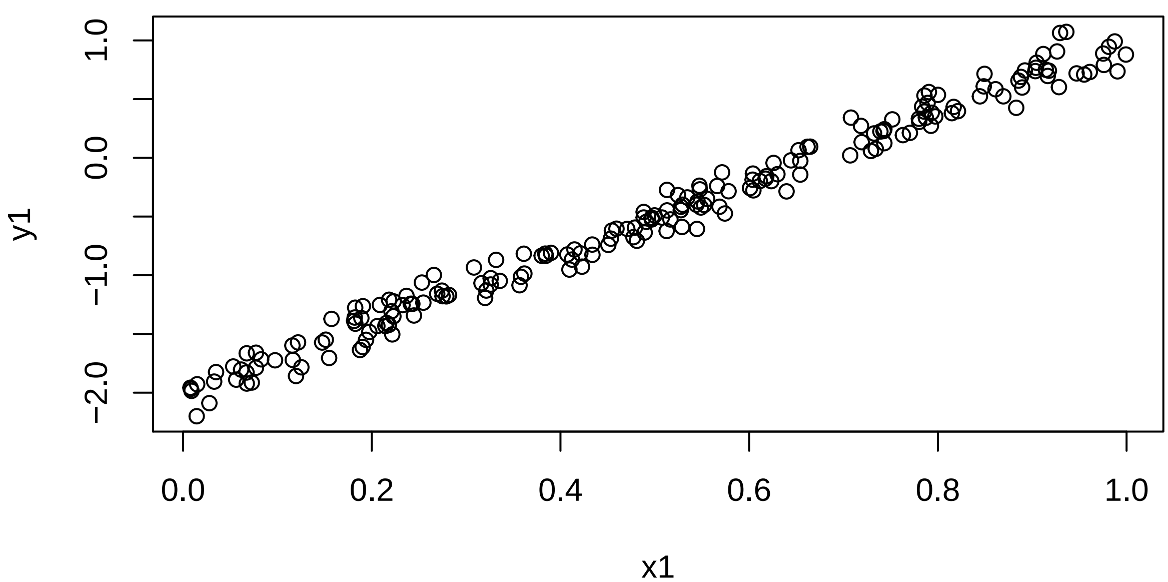
- How many harmonics make sense to include (maximum) when the period is in 24 steps?

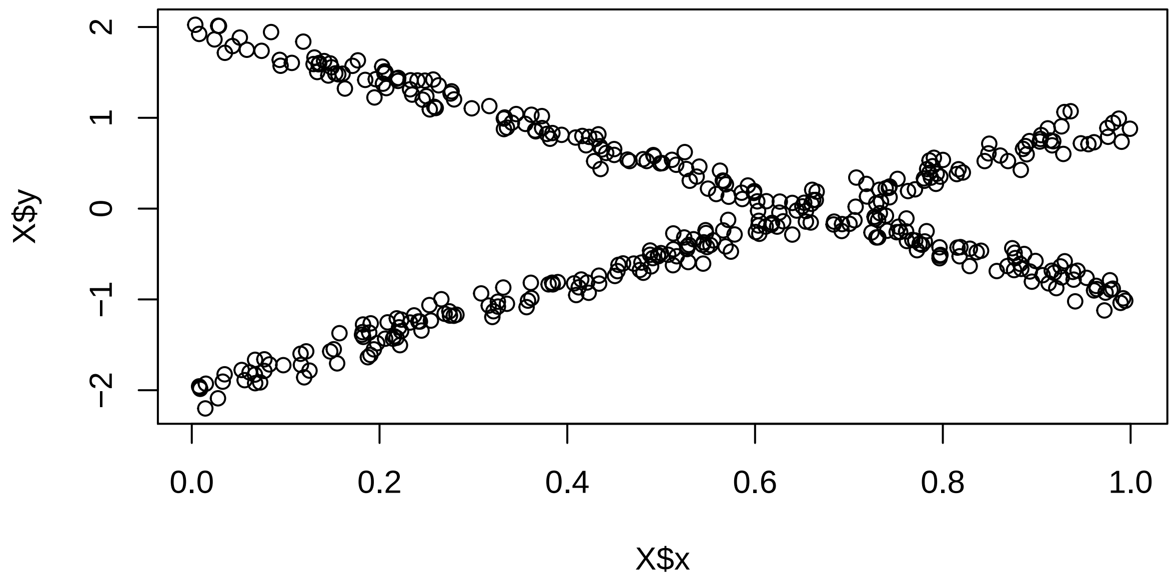
At least 3 harmonics are required to run the optimization. Even when adding more the optimization come to 3 pairs of sin() & cos() functions.

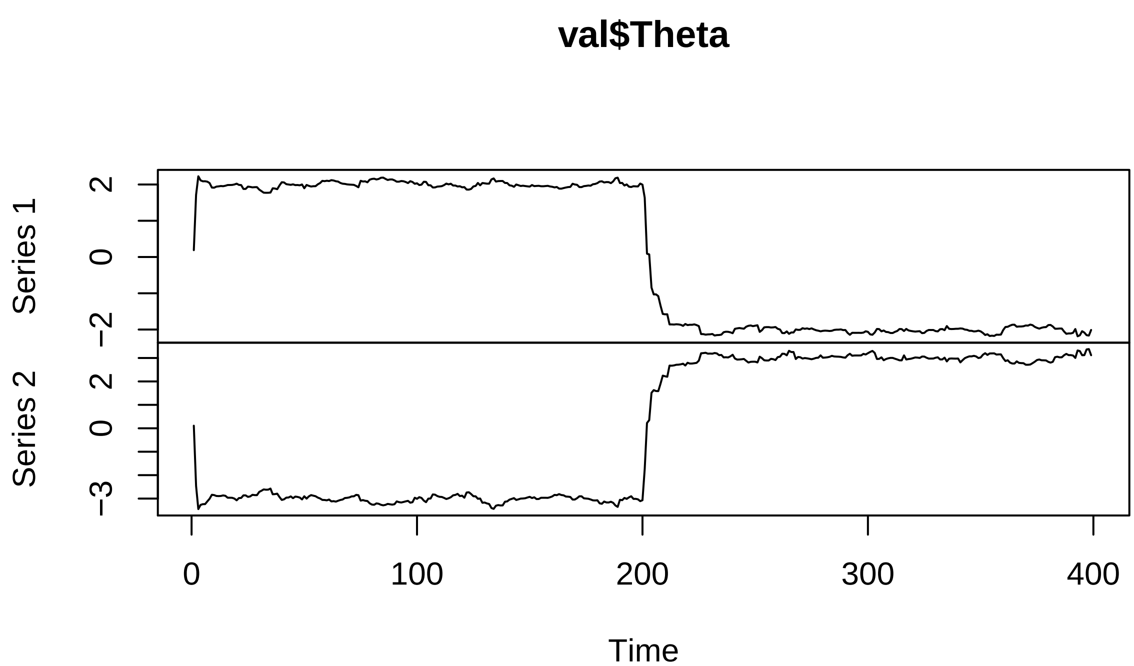


Q4 - Recursive least squares









- Does lm() estimate the parameters well on the combined data?

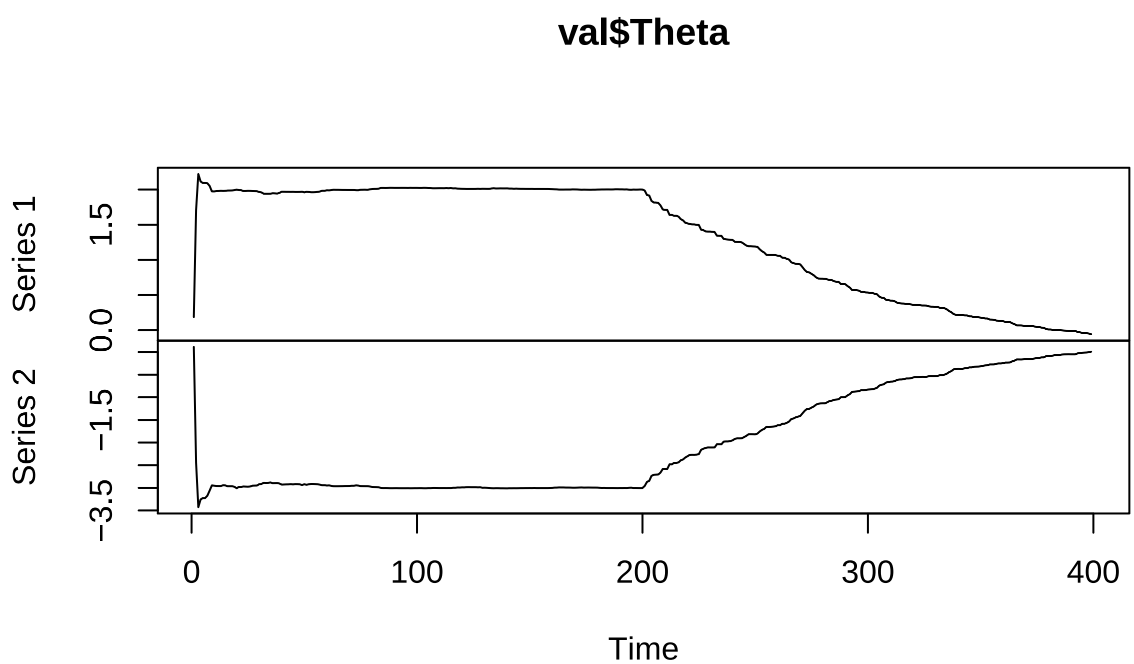
*No, **lm()** can estimate the coefficientes of the individual series but when they are combined it can not find the parameters since it is not anymore a line and just finds a middle value.*

- Does rls()?

Yes, using ***rls()*** *it is possible to estimate the parameters of the combined data.*

Try to change the forgetting factor λ.
* What happens when it is set to 1 compared to the results obtained from lm()?

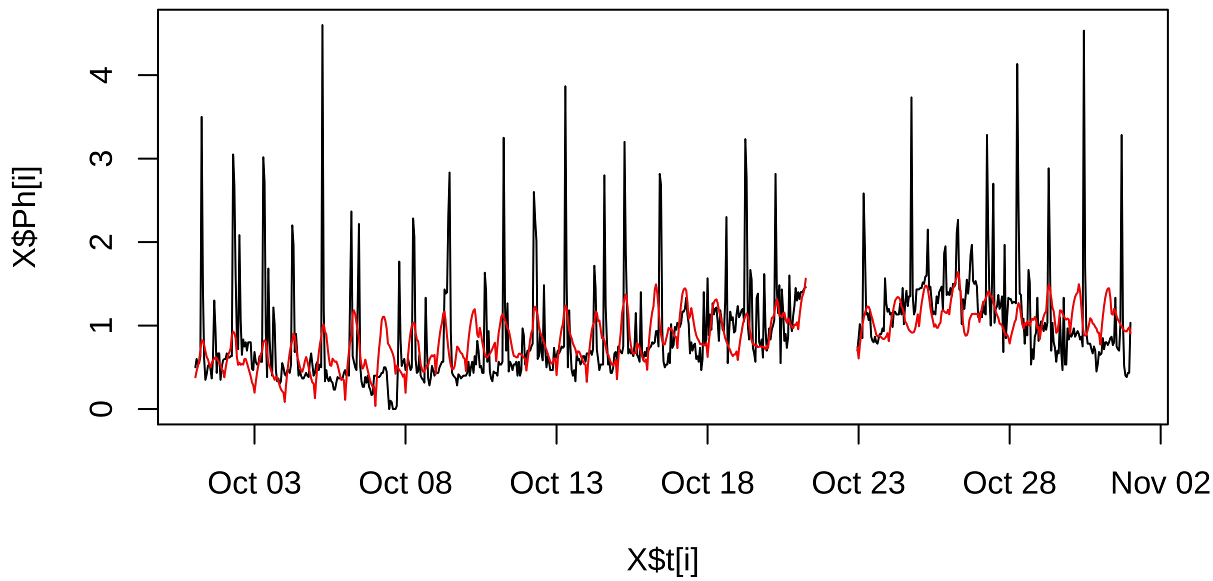
The forgetting factor tells how much weight the model give to older data, $\lambda = 0$ *means that the model "lives in the present" and ignores previous data;* $\lambda = 1$ *means that the model "remembers" everything. When setting* $\lambda = 1$ *the last coefficients are very close to those obtained by the simple* ***lm()***.

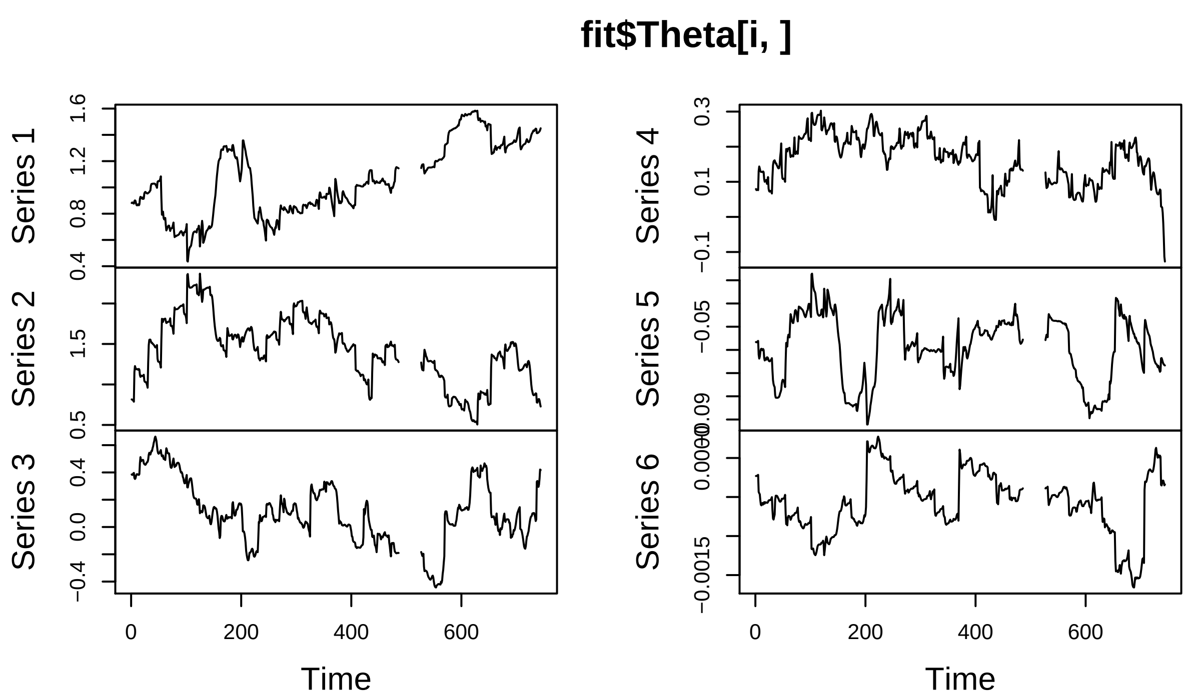


- Is there a trade-off between variance and bias (i.e. over- and under-fitting) related to λ?
  Yes, the forgetting coefficient has to be chosen in the right way otherwise the model will not be fitted properly.
  If the forgetting coefficient is too small (model "lives in the present"), then there is a problem of over-fitting because it will try to adjust to every new comming value, having a big Variance.
  On the other hand, a high forgetting coefficient will give under-fitting and the values will be biassed by very old observations.

Q5 - Load forecast with RLS

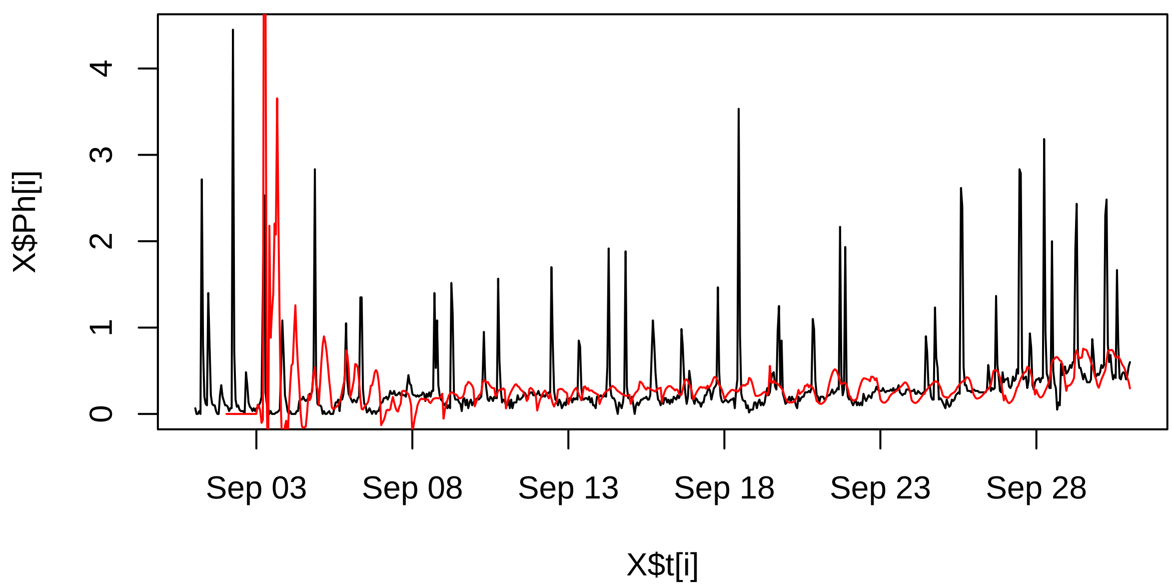
Now we will use RLS for fitting the coefficients, hence they can change over time.

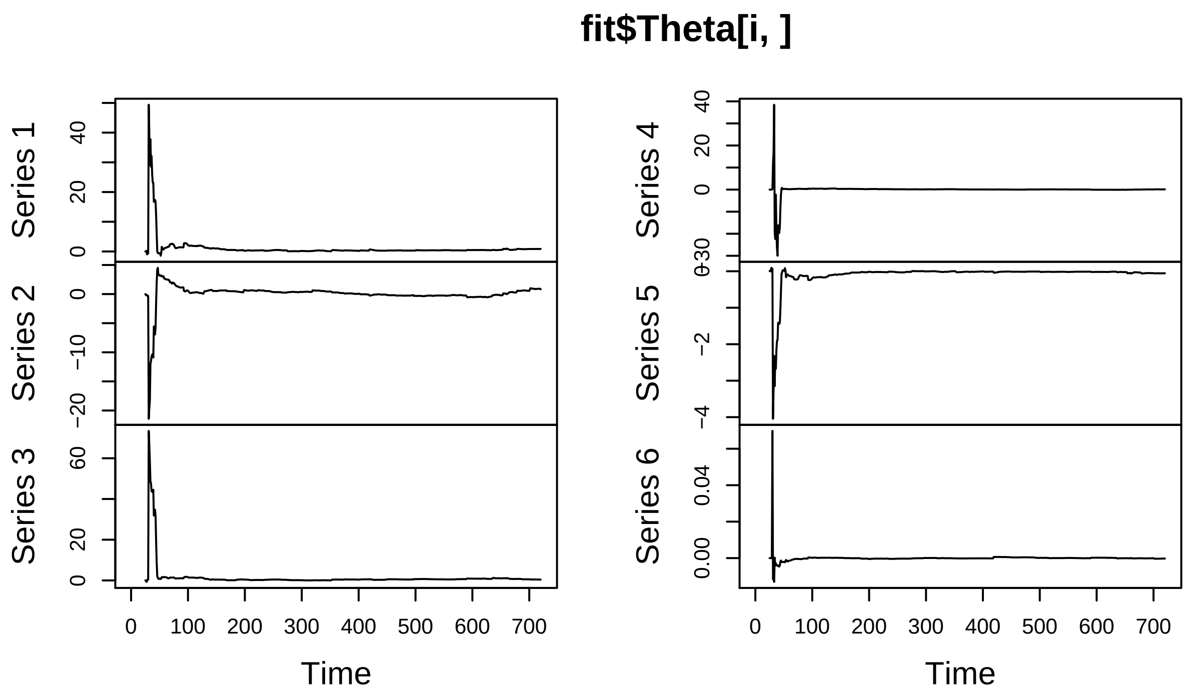




- How about the forecasts, do they look fine?
  No, the load forecast (in red) does not seem to follow properly the actual load from the test set.
- The tracked coefficients (the βs kept in θ), do they change?
  The coefficients seem to be fluctuating too much.
- What was λ set to? it that optimal? λ *was set to 0.99 but it doesn't seem to be optimal*

Run the next part plotting the first month of the training set:

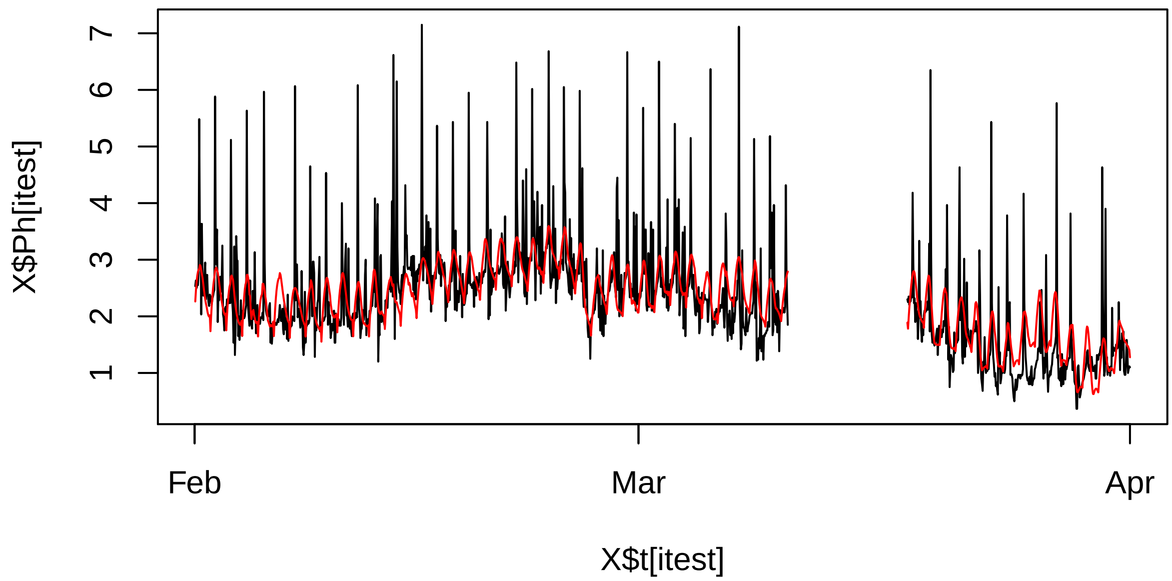


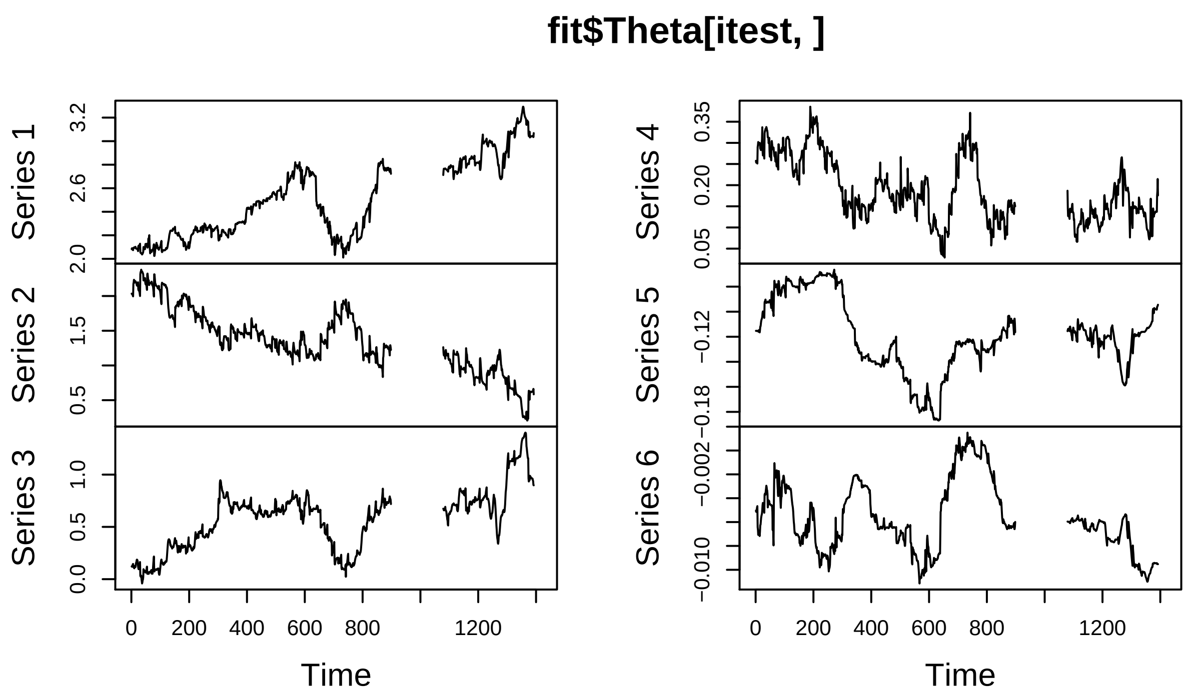


- Are the forecasts good the first week?
  No, the forecast follow somehow the trend of the load but not the peaks and there seems to be a lag between the observed values and the forecasted ones.

- What about the coefficients?
  During the first period the coefficients vary a lot (see the y-axis of hte Theta-graph). This is because the model has not enough "history/memory" to properly predict.

Now, since the forecasts are poor until the coefficients are tracked, then make a "burn-in period", which simply means that a period in the beginning of the training set is left out in the score evaluation.



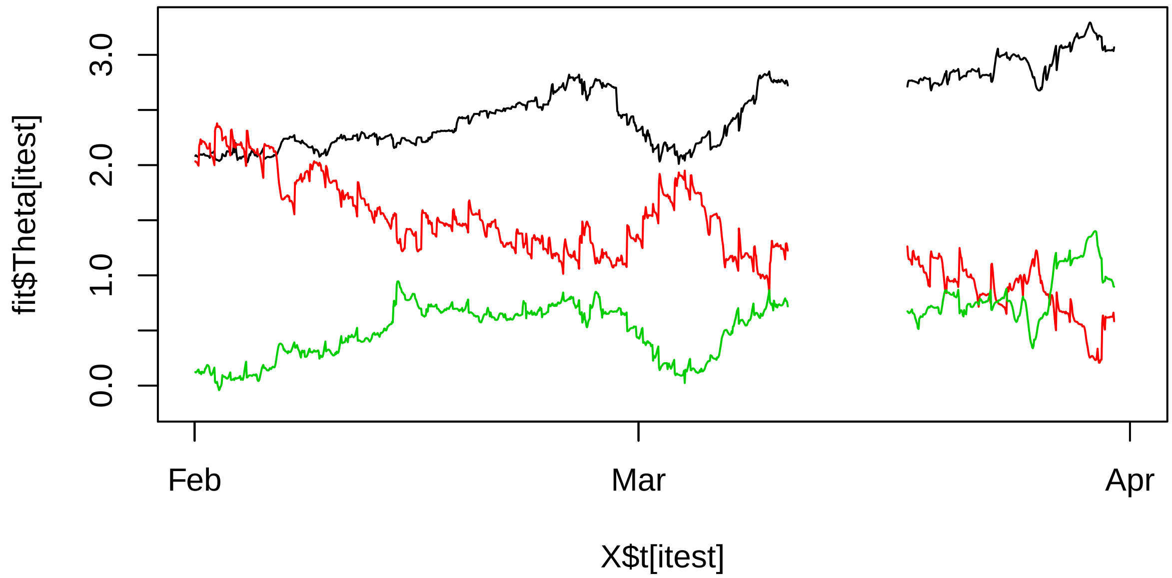


- Are the forecasts good the first week?

By removing the first period (burn-in period) and optimizing the parameters the forecast became better.

- What about the coefficients, did they change?

The coefficients are now more stable now and the range in which they vary is much smaller

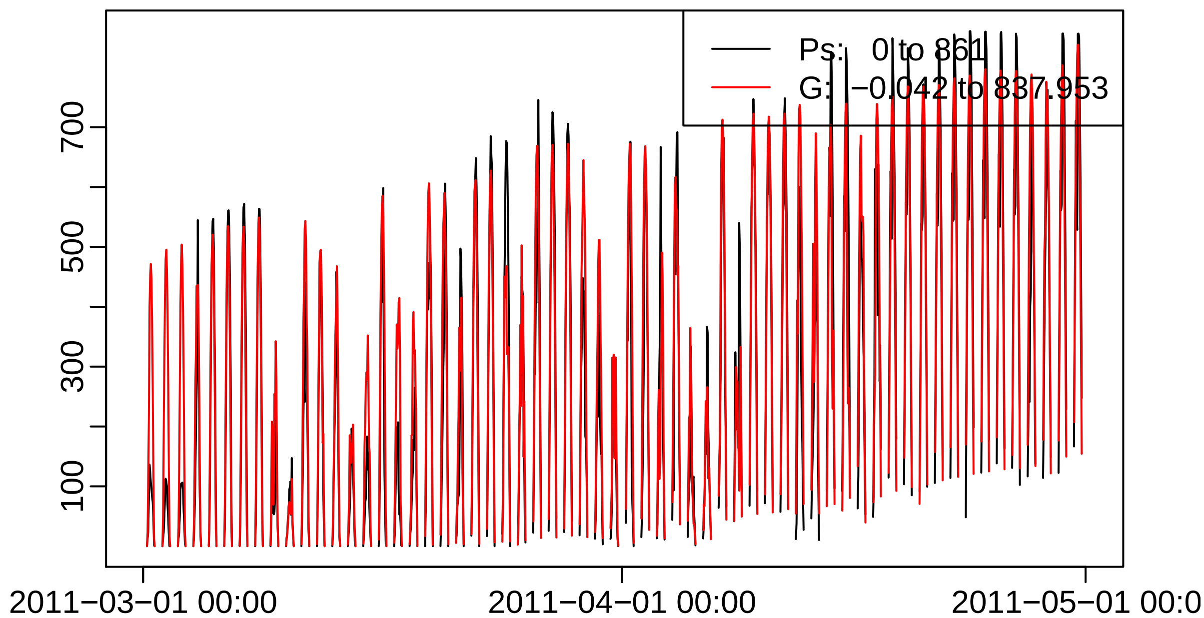


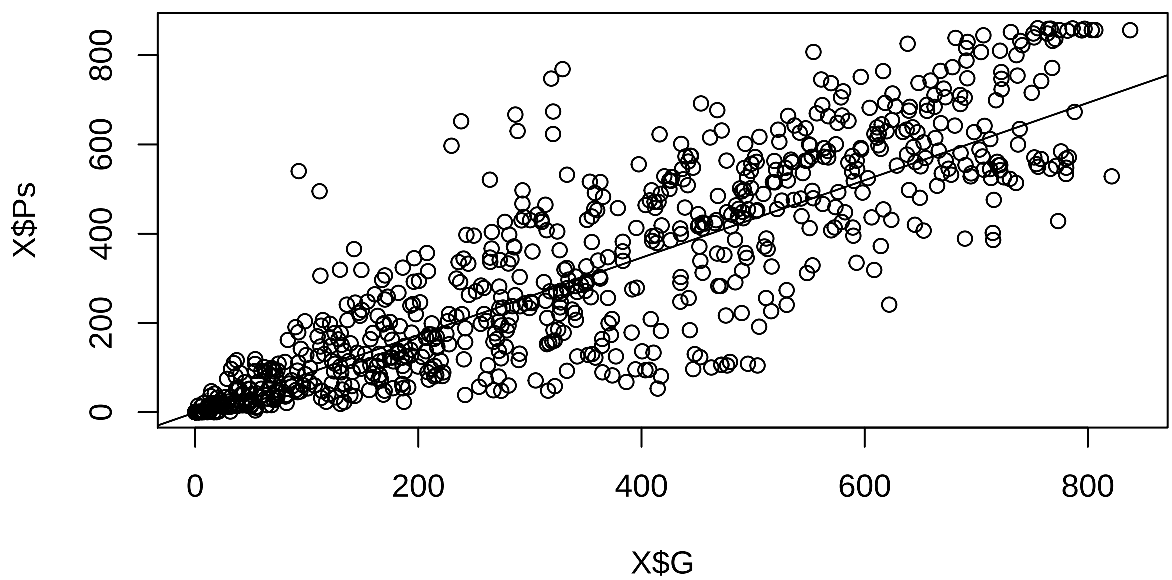
- Did the forecast improve? *Yes, there was an improvement in the forecast when using RLS. The RMSE from 0.789 to 0.73 compared to using the tuned low-pass filter.*
- Does the coefficients change over time? *Yes, the coefficients are now changing over time perhaps in a seasonal way.*

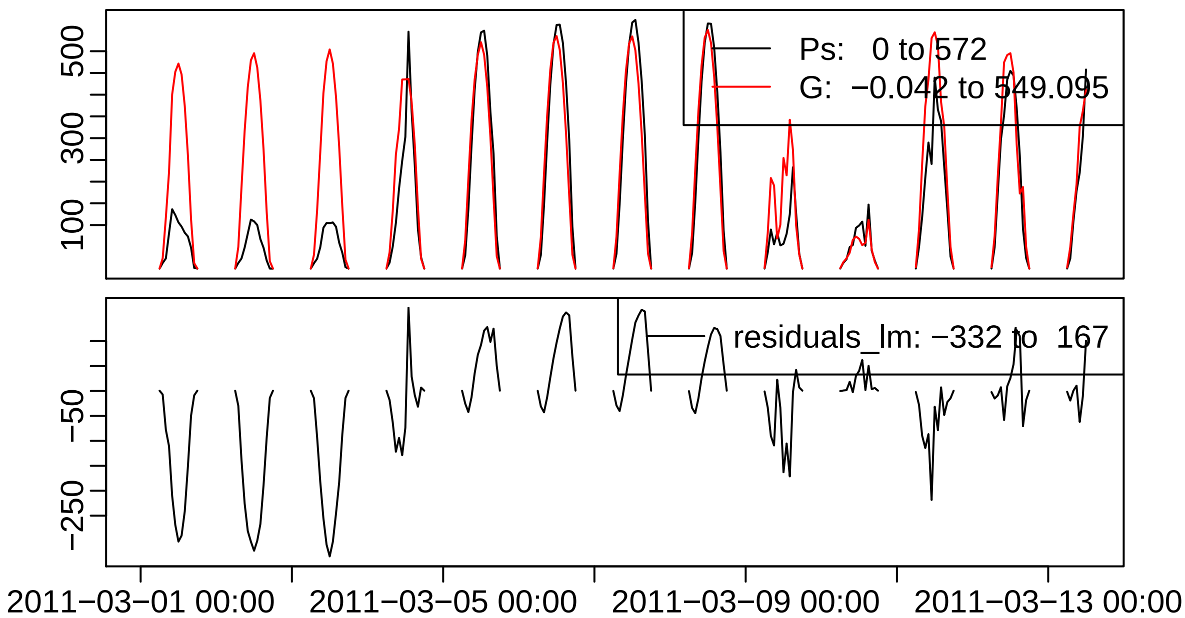
Q6 - Solar forecasting

Now we can "easily" find a model which is useful for forecasting solar, e.g. the power generation on a PV panel. In this exercise, we will actually just use the observed global radiation as the solar power, hence this is somewhat a little bit simplified case. However, these observations contain quite a few deviations: **shadowing in the late morning hours from a chimney, some tilt of the sensor and also some saturation.**

Simple linear regression model is fitted







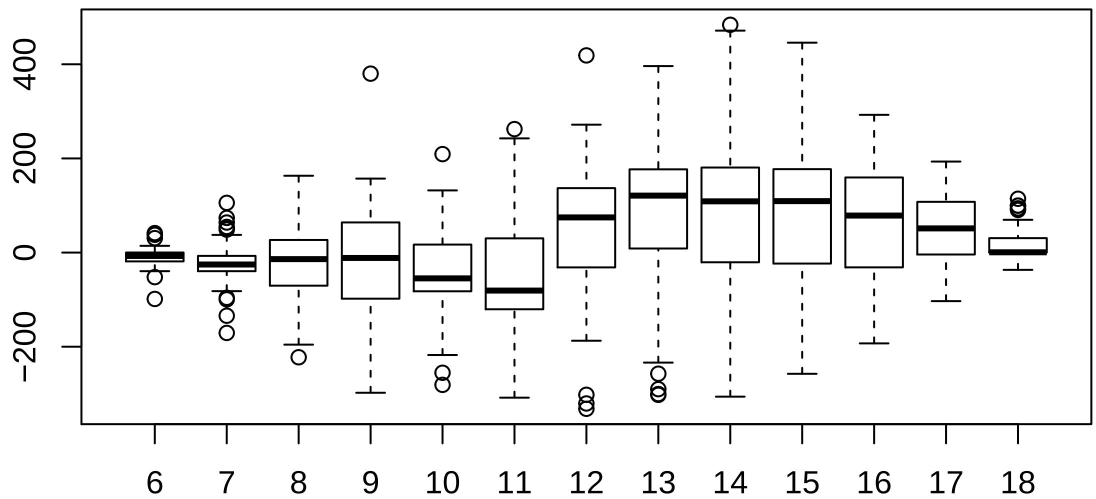
- Do the forecasts seem to be good?

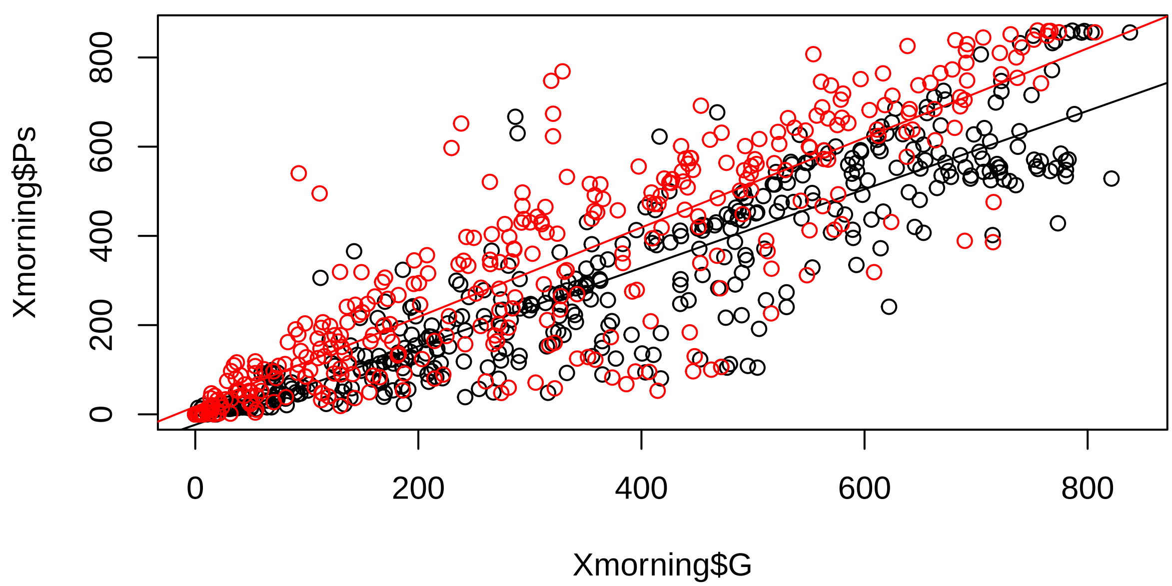
The forecast seems to match in some parts but it is not really good.

- Can you spot any systematic patterns?

There are some periods (several days in a row) where the prediction is above the observed value, and some days where it is the other way around.

Explore the residuals in the next part:





- Do you find any systematic patterns?

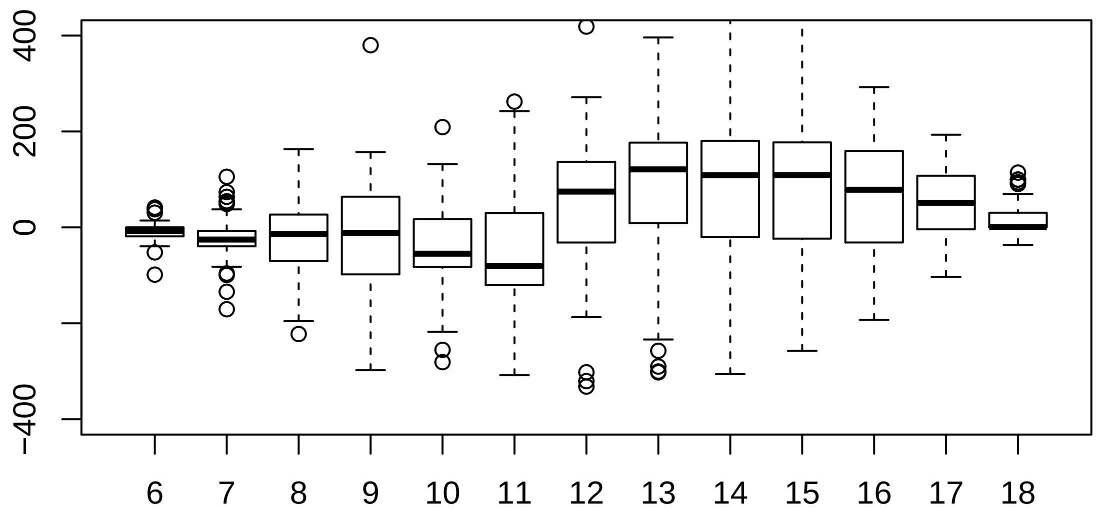
Yes, specially in the middle of the day the spread is much bigger. In the mornings the residuals are mainly below 0, while in the afternoon they are greater than 0.

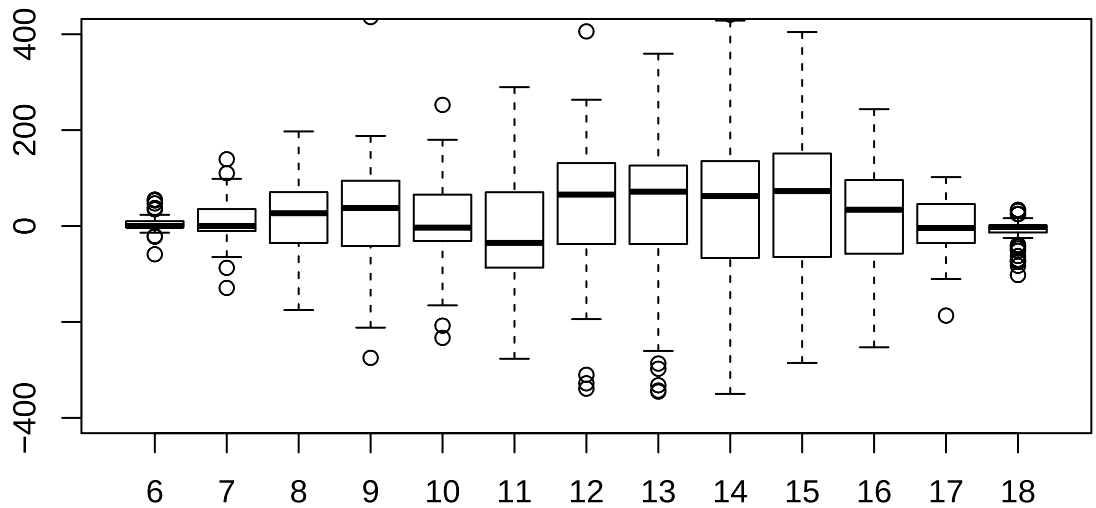
- What can cause the found differences in the relation between NWP global radiation (G_{nwp}) and the observed solar power (i.e. observed global radiation) P_s?

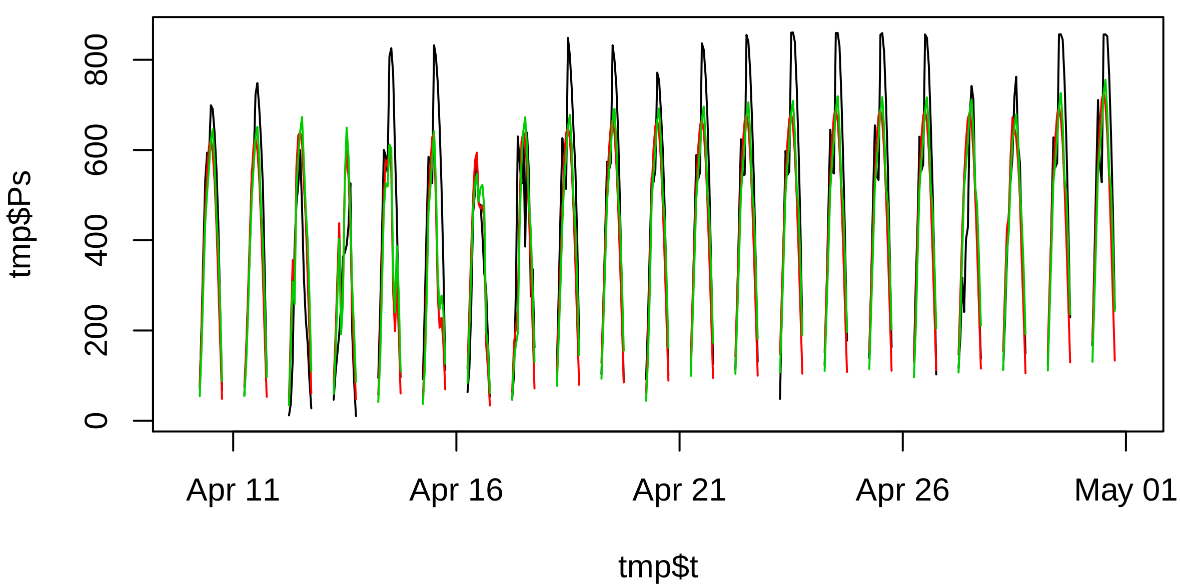
These differences could be causd by the shadding of the chimmeny that appears in the late morning, differences with the orientations/tilt of the sensor.

Now, define a model the relation between the solar power and the global radiation NWP is conditional on the time of day. Fit it, using base splines and lm() and rls(), and finally with a kernel model.

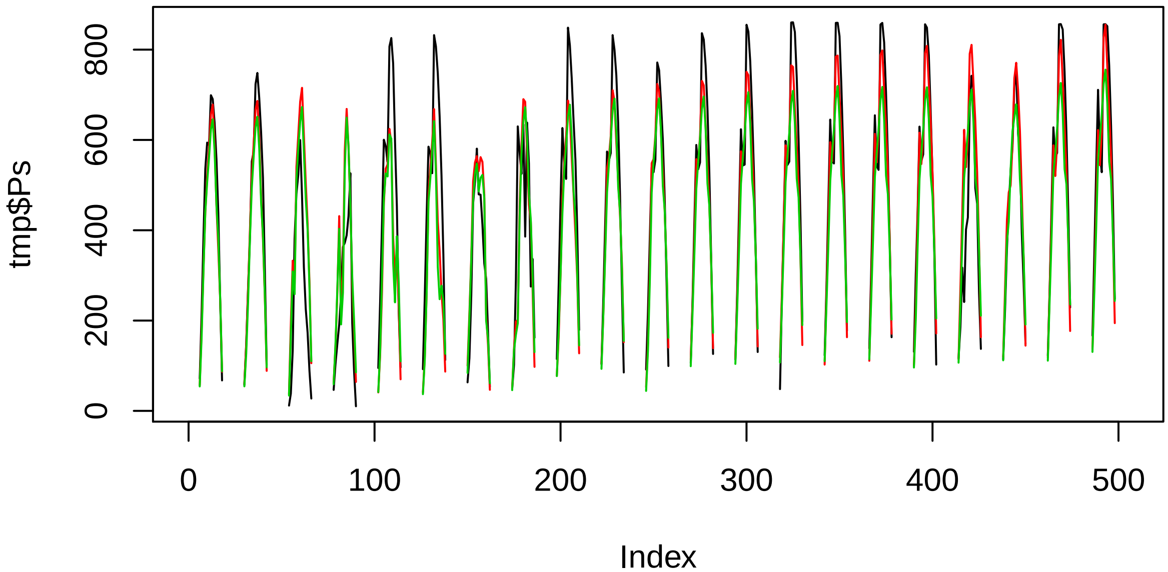
Base splines



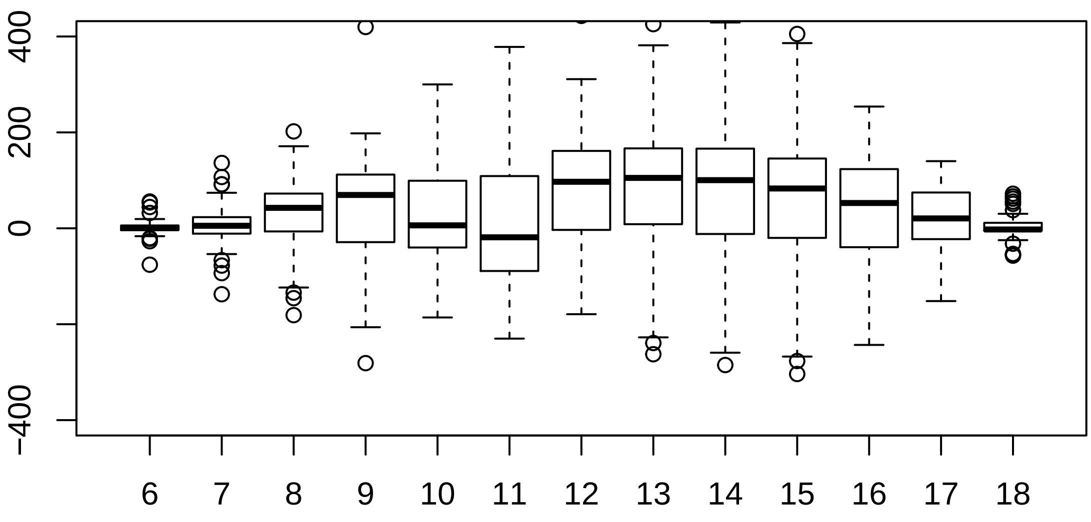

400
200
0
−400
6
7
8
9
10
11
12
13
14
15
16
17
18

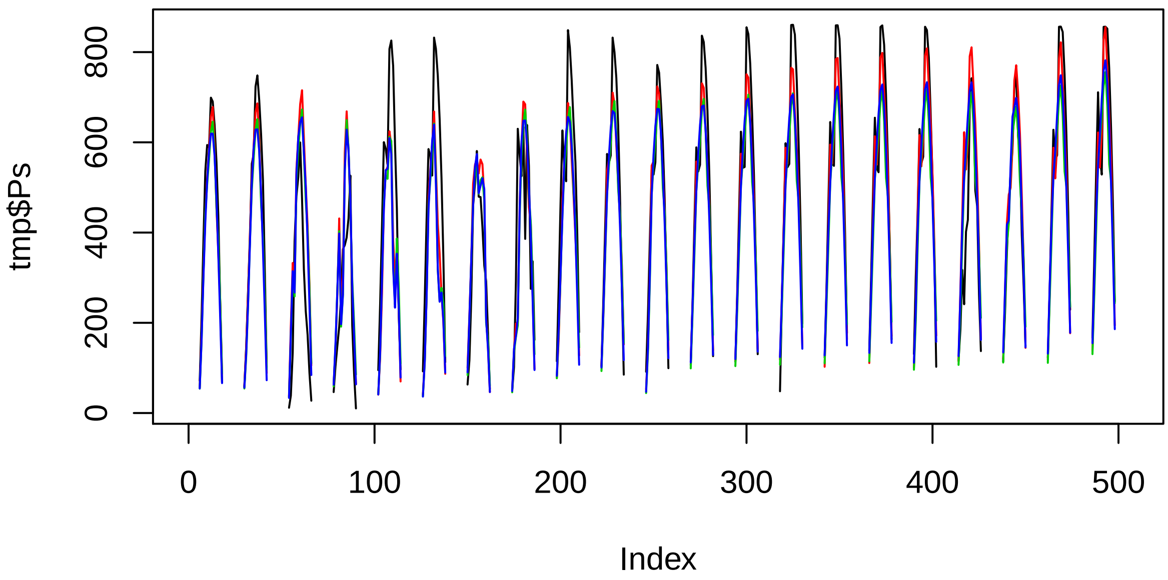

tmp$Ps
0
200
400
600
800
Apr 11
Apr 16
Apr 21
Apr 26
May 01
tmp$t

Base spline model with rls



Kernel model






```
##                  Model     RMSE
## 1               Kernel 118.2419
## 2  base spline and lm 120.2910
## 3 base spline and rls 107.0503
## 4                   lm 137.2229
```

- How can you decide which model is better?

Based on the RMSE and the plots, "Base spline and RLS" is the model that predicts better the solar power.

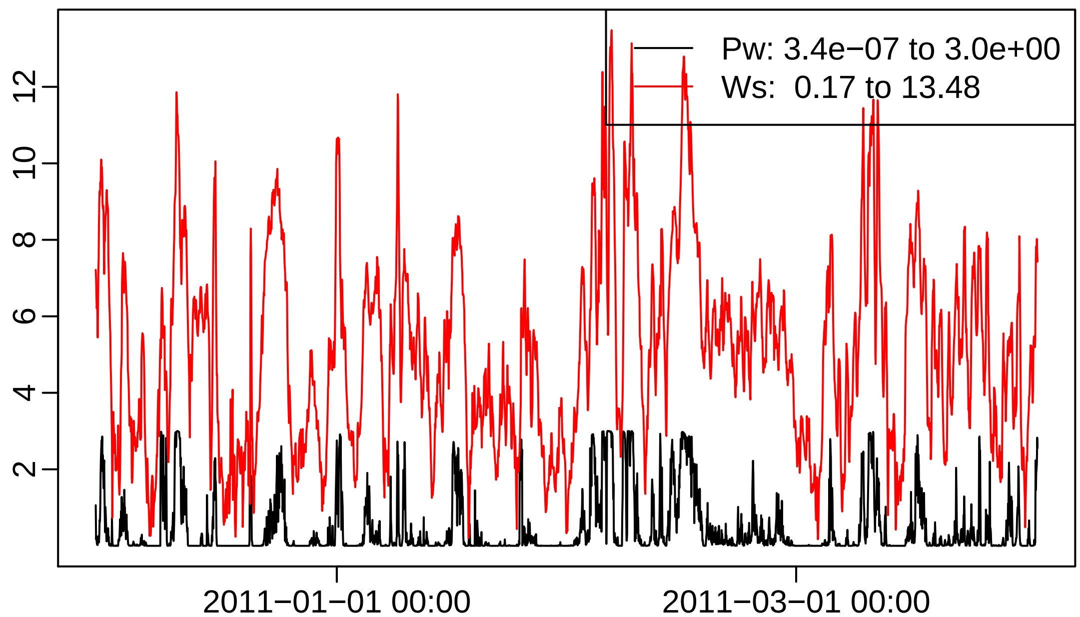
- Do you find differences between the prediction performance of the models?

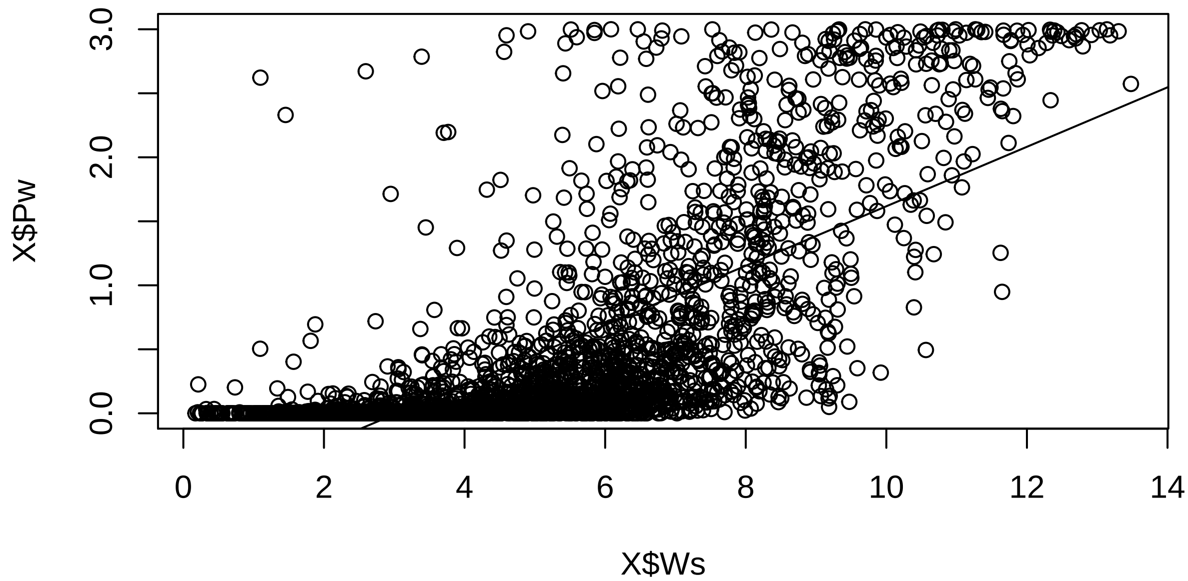
Yes, specially when applying only the Linear Model (lm) because it lacks the ability to adapt the variance with time

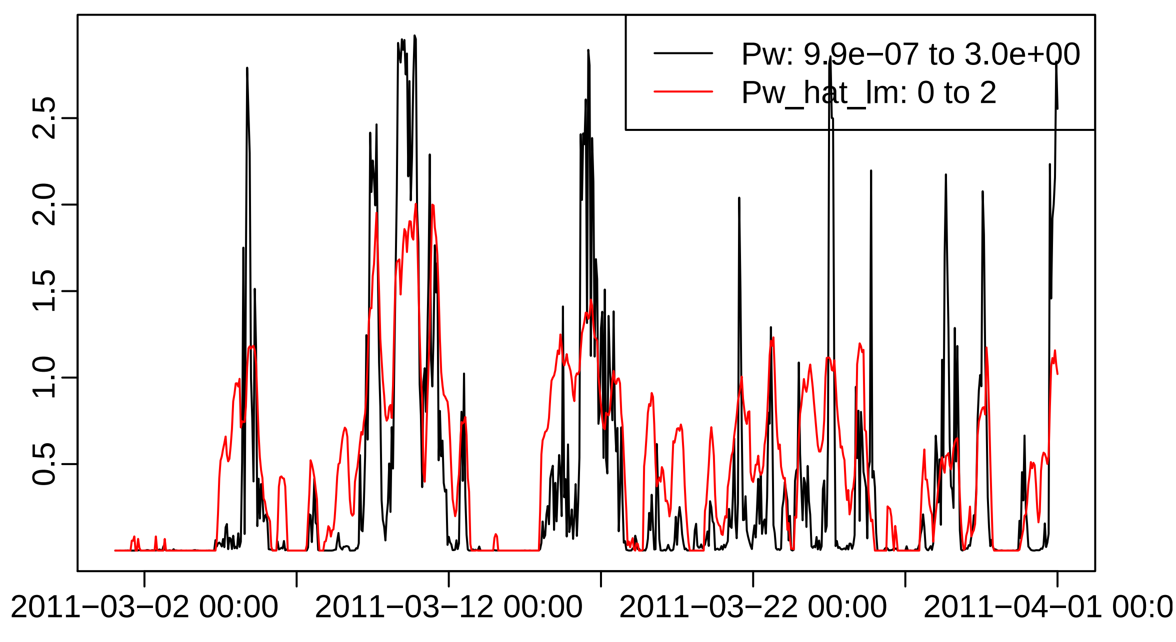
Q7 - Wind forecasting

These are not real wind power measurements. The “wind power” Pw, which is used, is actually the observed wind speed put through a simple power curve function. Hence this is a “simplified” wind power, which has the basic properties of a “real” wind power – remember “real” wind power signals are potentially quite different depending on the wind turbine and wind farm properties, and surroundings etc. Hence, the signals used holds the basic properties of a wind power measurement.

Linear regression model





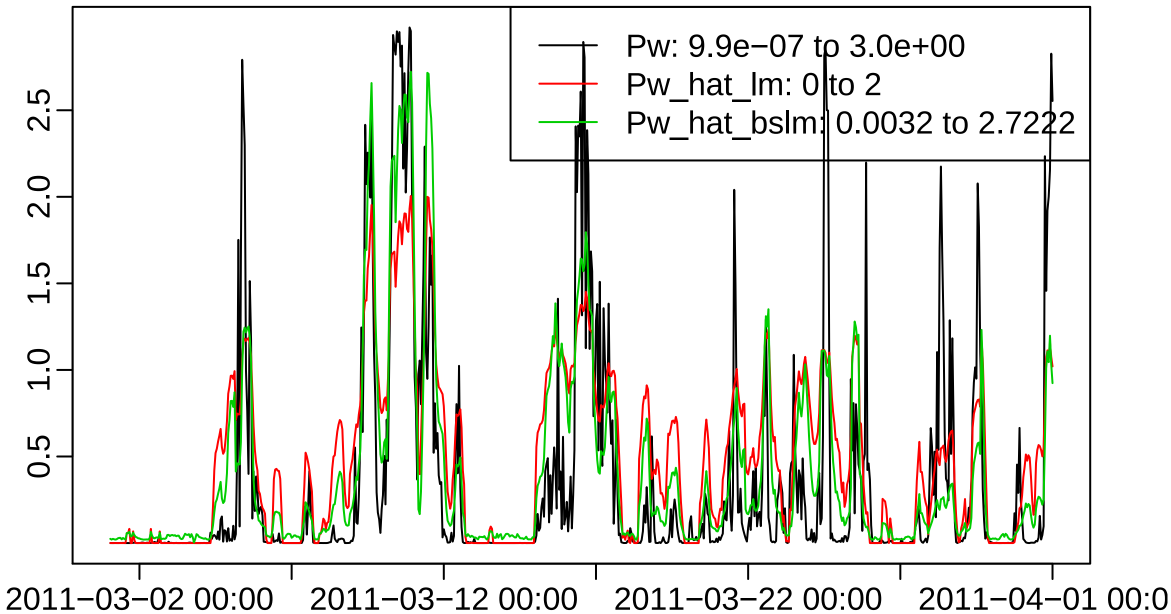


Look at the scatter plot of the wind power P_w vs. wind speed NWP W_s (`plot(X$Ws, X$Pw)`) with the linear model fit (`abline(fit)`). * Can you conclude that a linear model is suitable?

No, a linear model is definitively not suitable in this case since the data doesn't follow a linear path (looks *more like a logistic regression)*

Spline model

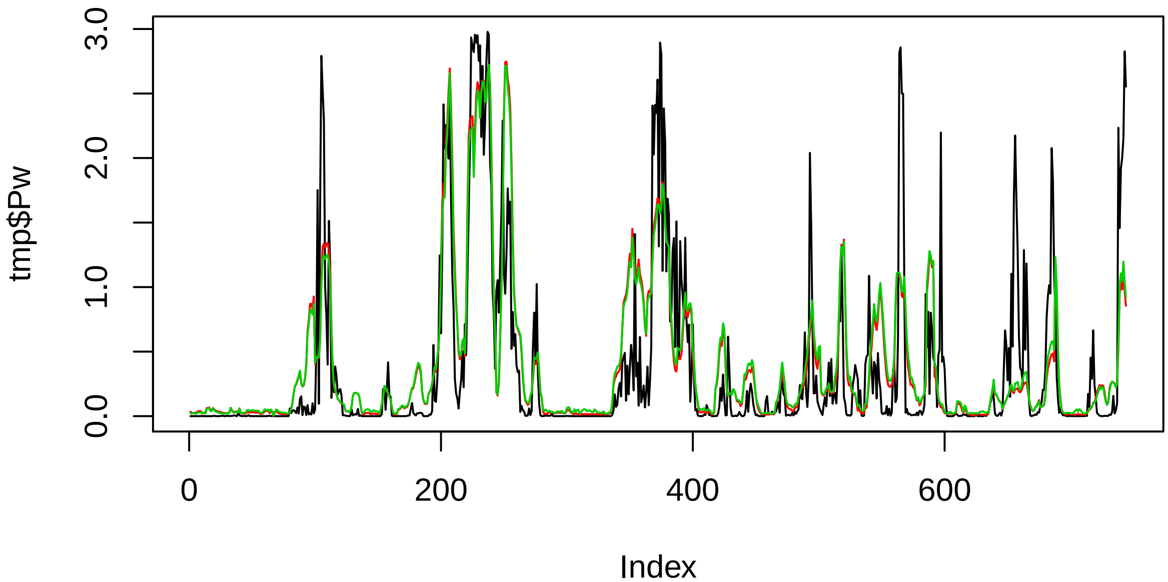
Making the model non-linear in the wind speed



- Are the forecast improving when you apply a base spline model taking into account a non-linear functional relationship between W_s and P_w?

Yes, based on the graph of the forecasts and the actual wind power (P_W), the base spline model has a better fit than the linear regression model. Also, comparing the RMSE of 0.481 for the linear regression model to 0.43, there was an imporvement by uisng a non-linear relationship.

Base spline model with rls



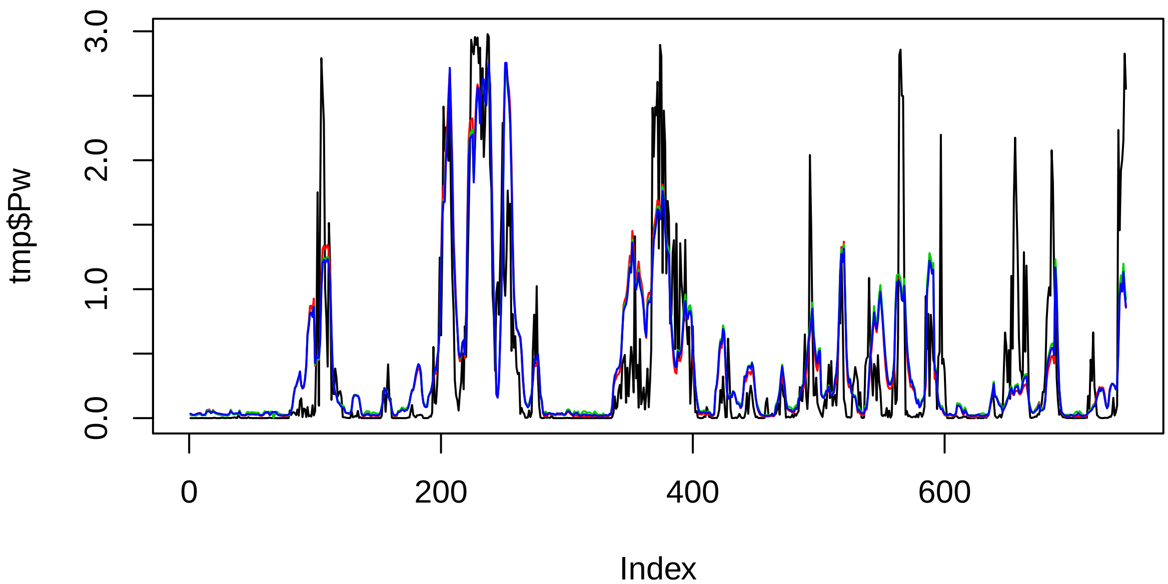
- Do we achieve improvements with RLS over LM?

There was no improvement when comparing the RMSE of 0.43 agaisnt the RMSE of the rls model 0.435

- If not, can you give some explanation? (i.e. somehow weird, since RLS with $\lambda = 1$ give the same result as LM, right!?)... what was the optimal λ found using the training set?

The lack of improvements could come because the minimum lambda encountered thorugh the optimisation (0.999 is very close to 1 which gives the same results to Lm, so no big improvement is achieved.

Kernel model



- Finally, with the kernel model, do we achieve improvements?

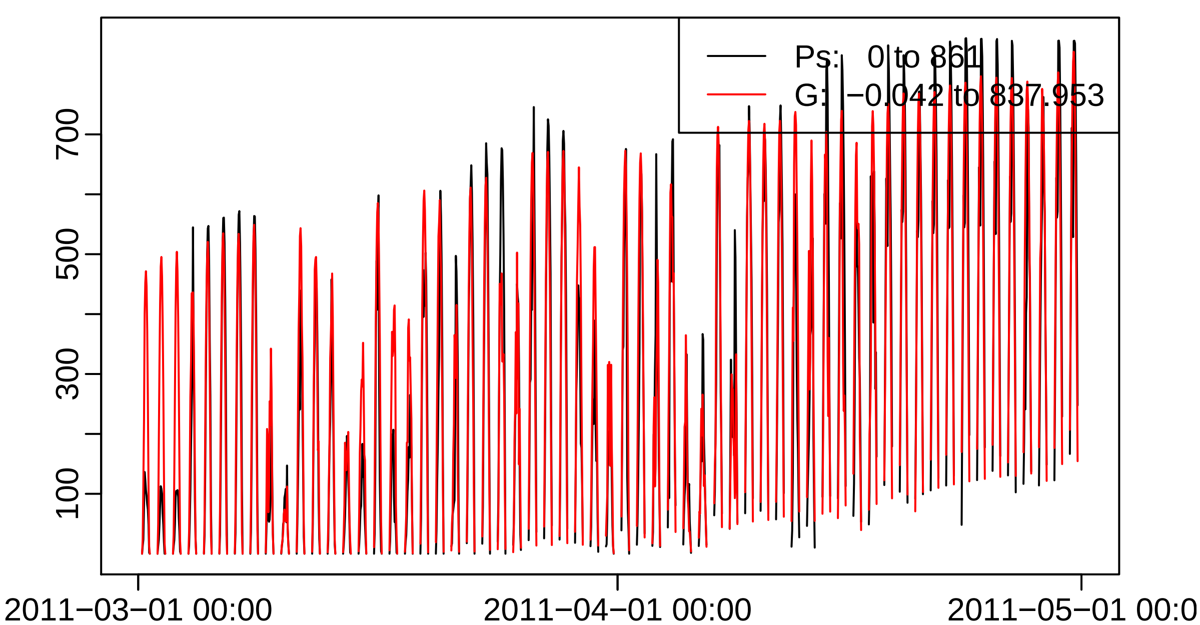
The kernel model gives a very similar fit as the previous models but if we just consider the RMSE then the "Base spline with lm" is still the best option.

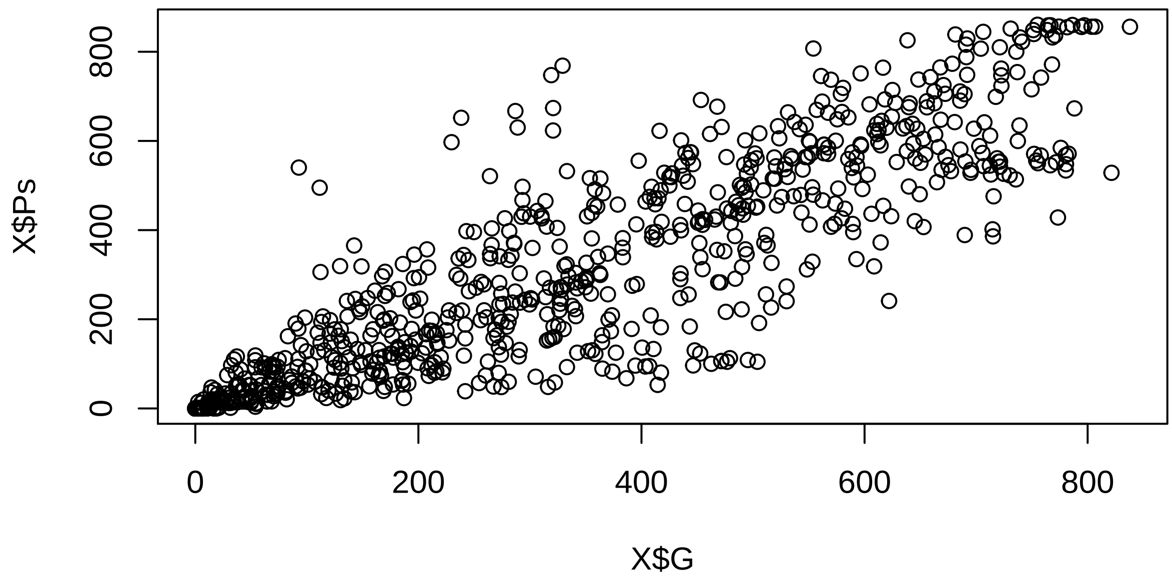
So, this was a very "simplified" example, when forecasting "real" wind power of, say, a huge wind farm, then there are many things to take into account. First, surely a conditional dependence of wind direction is important, and further, a lot of information about the operational status is very important to take into account.

Q8 - Probabilistic forecasting

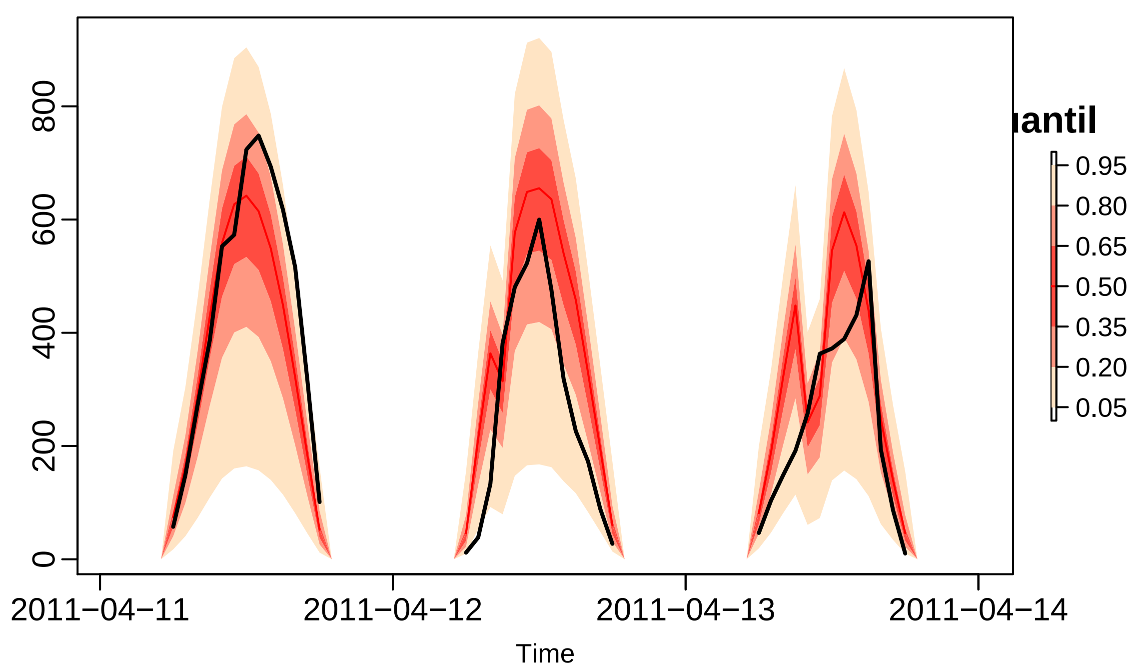
Finally, an example of how a probabilistic forecasting model can be setup is presented. It is very simple: replace linear regression lm() with quantile regression rq(), and replace the RMSE score function with Continuous Ranked Probability Score (CRPS).

> Every possible future has an assigned probability when doing probabilistic forecast. These forescasts however at not equally accurate and there is a need for evaluating them. There are simple metrics that asses the accurary such as MAE (Mean Absolute Error) or MAPE (Mean Absolute Percentage Error) but this are not suitable for probabilistic forecasts.
> The Continuous Ranked Probability Score (CRPS) generalizes the MAE to the case of probabilistic forecasts. It compares a full distribution with the observation, where both are represented as cumulative distribution functions (cdfs). It is sensitive to the entire permissible range of the parameter of interest. It can be interpreted as an integral over all possible scores. For the case of a deterministic forecast, the CRPS is equal to the mean absolute error (MAE) and, therefore, has a clear interpretation (Gneiting and Raftery [2007], Hersbach [2000], Brown [1974]).

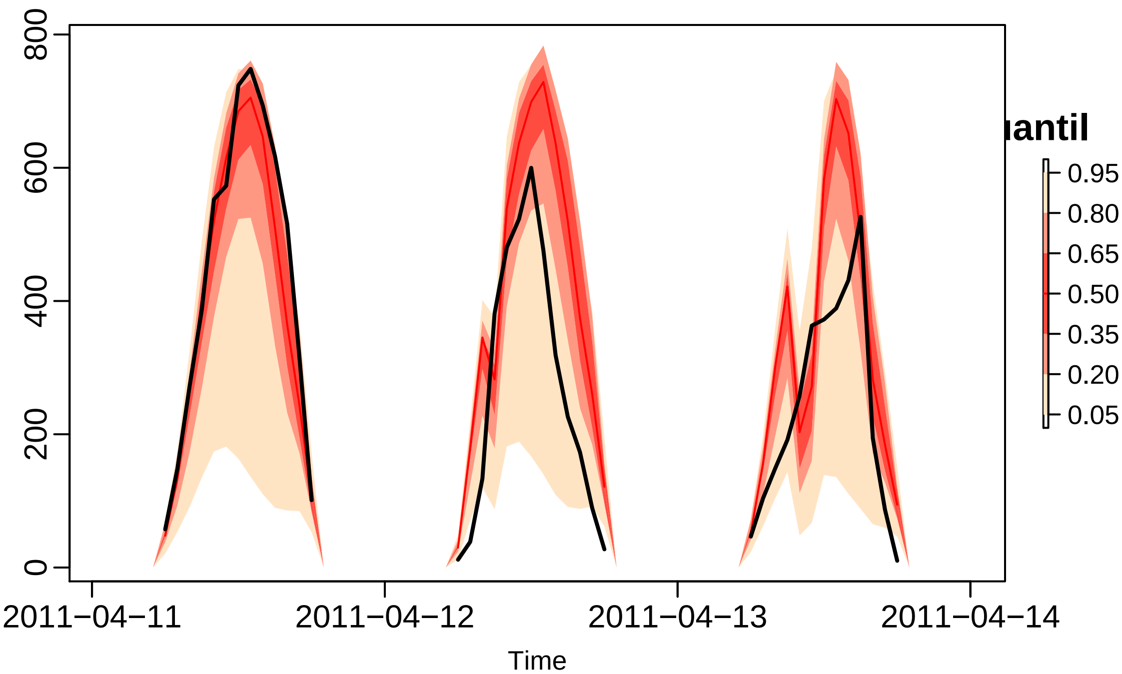




Quantile Regression model



Base spline model with Quantile Regression



- Do we gain in predictive performace using the base spline model over the linear model? *Yes, looking at the graphs we can see that the test-values fall more into the quantiles of the base spline model compared to the linear model. Also the CRPS improves from 79.678 in the linear model to 68.428 using base splines.*

References

JJ Allaire, Yihui Xie, Jonathan McPherson, Javier Luraschi, Kevin Ushey, Aron Atkins, Hadley Wickham, Joe Cheng, and Winston Chang. *rmarkdown: Dynamic Documents for R*, 2018. URL https://CRAN.R-project.org/package=rmarkdown. R package version 1.10.

Peder Bacher and Henrik Madsen. Experiments and data for building energy performance analysis: Financed by the danish electricity saving trust. 2010.

Peder Bacher and Henrik Madsen. Identifying suitable models for the heat dynamics of buildings. *Energy and Buildings*, 43(7):1511–1522, 2011.

Thomas A Brown. Admissible scoring systems for continuous distributions. 1974.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Keith R Godfrey. Correlation methods. In *System Identification*, pages 527–534. Elsevier, 1981.

Hans Hersbach. Decomposition of the continuous ranked probability score for ensemble prediction systems. *Weather and Forecasting*, 15(5):559–570, 2000.

Rune Juhl. *ctsmr: CTSM for R*, 2018. R package version 0.6.17.

Henrik Madsen. *Time series analysis*. Chapman and Hall/CRC, 2007.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2018. URL https://www.R-project.org/.

Hadley Wickham. *tidyverse: Easily Install and Load the 'Tidyverse'*, 2017. URL https://CRAN.R-project.org/package=tidyverse. R package version 1.2.1.

Yihui Xie. *knitr: A General-Purpose Package for Dynamic Report Generation in R*, 2018. URL https://CRAN.R-project.org/package=knitr. R package version 1.20.