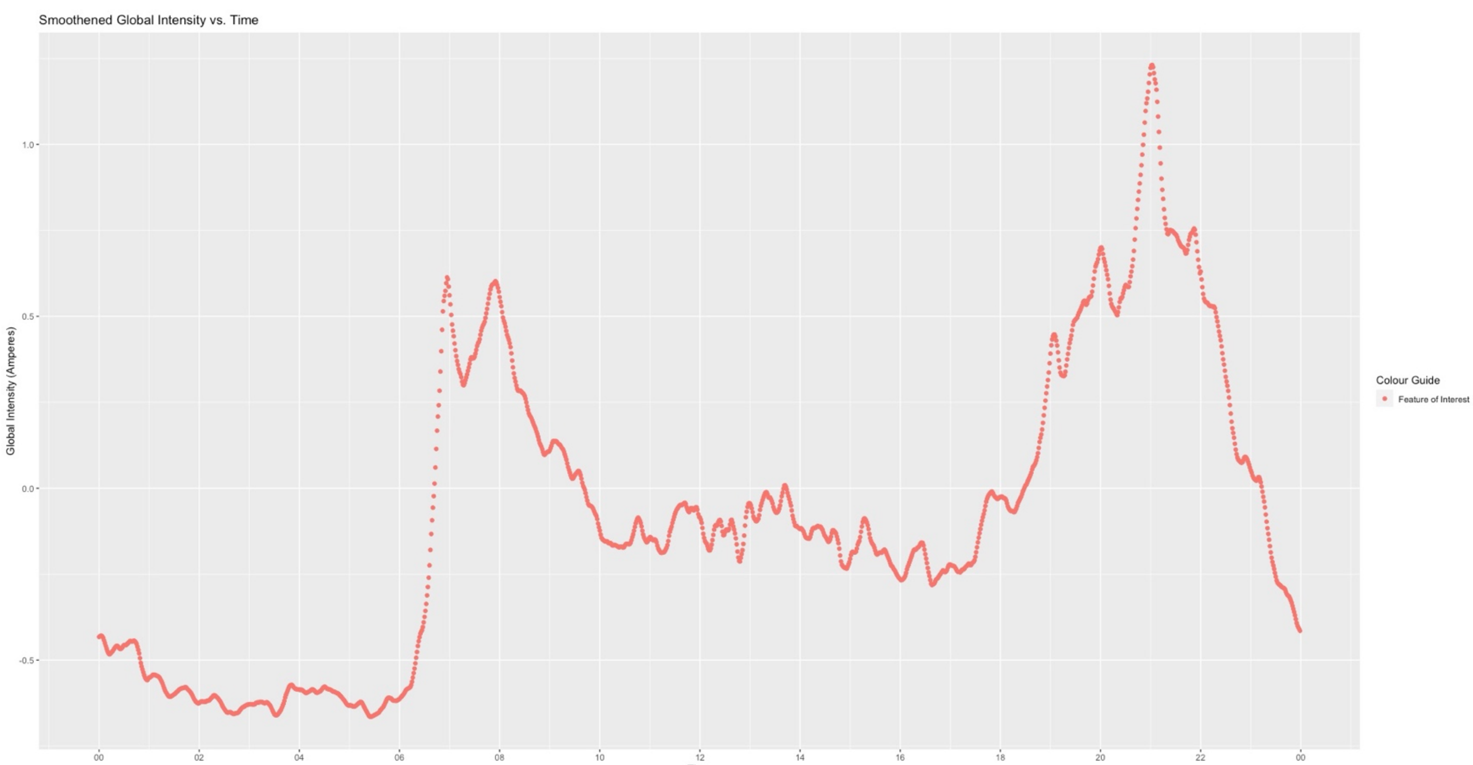
1. **Scale the dataset provided, e.g. by means of the R command scale(). For one of the response variables, Global_active_power, Global_reactive_power or Global_intensity, determine a time window for a specific weekday that shows a clearly recognizable electricity consumption pattern over a time period of not less than 120 and not more than 240 minutes. Extract the same time window for each week of the dataset and concatenate the extracted time windows to build a dataset for the training of HMMs.**

To accomplish step one, we started as recommended by standardizing the features of the given data set via scale.

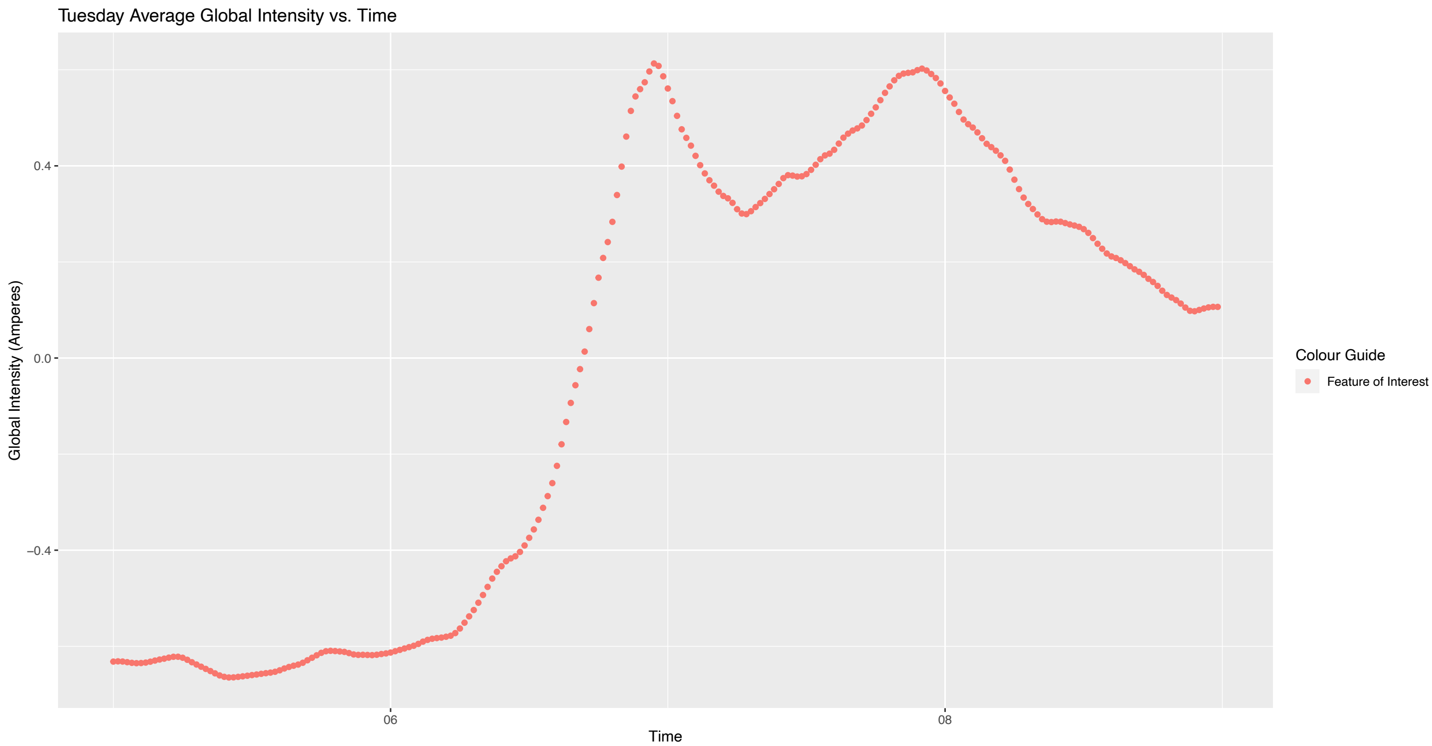
For the choice of response variable, we decided to use Global_intensity.

To determine a time window for the specific weekday that showed a clearly recognizable electricity consumption pattern, we first computed the average smoothened week for each week in 2007 by smoothening the Global_intensity data for each week using the moving average, with a time window of 10 consecutive observations. We then computed the average smoothened week by finding the averaged value for time for all times for one whole week. We then chose a day, Tuesday, as the day of choice, and plotted the average Tuesday to try to identify an interesting and definable time window:



Here we noticed two distinct interesting time periods: 05:00 – 12:00 and 16:00 – 24:00. Noting the fact that we were to analyze a time period between 120 – 240 minutes, we chose 05:00 – 09:00, as it appeared to have a distinct and interesting rise in usage, coupled with a “calm” section that existed prior to 06:00.

See the following page for a graph of the section we chose.



2007 Tuesday Average Global Intensity vs. Time for the time window 05:00 – 09:00.

2. **Use your training dataset for training a number of univariate HMMs that each have a different number of states across a range from not less than 3 states to not more than 16 states. For each HMM, compute the log-likelihood measure on the training dataset. In addition, compute the Bayesian information criterion1, or BIC, as a measure of the complexity of your model. The goal is to find the intercept of the two plots for log-likelihood and BIC values respectively so as to determine the best model (avoiding overfitting). You may not need to train HMMs for each and every number of states within the range by making smart choices.**

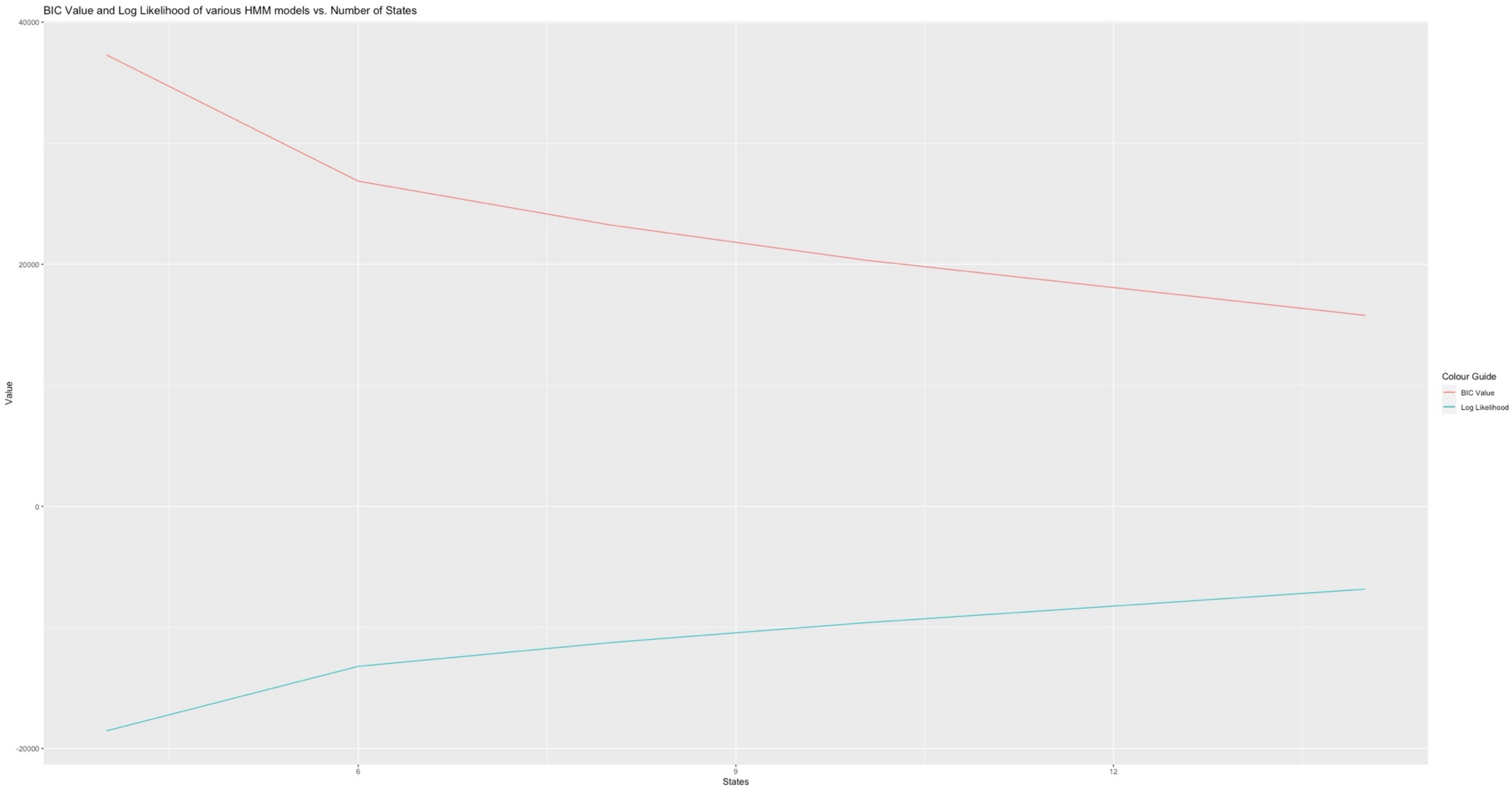
Here we trained 5 different models, with 4, 6, 8, 10, and 14 states, respectively. Our objective was to maximize log likelihood and minimize the BIC value, to find the optimal model.

Some comments:

We had some trouble getting models with > 10 states to converge. Between, 10 and 16 states, only models with 10 and 14 found convergence. We made many attempts using different seeds, as was recommended by Professor Amir Shahir, and were able to get a 14 state HMM model to converge, but we had no luck otherwise.

We predicted that the ideal number of states for the model would be around 11. This is because from 05:00 to 06:00 we have relative calm, and these subtle gradients might account for two states. From 06:00 – 07:00 we have a sweeping rise in usage, with two subtle transitions which we thought may account for three states. This is followed by a peak (one state), a quick decline (another state), then a more moderate build (another state), and one more moderate peak (another state), followed by a final decline (two states, as there are some transitions to account for).

Please see the following page for a graphical representation.



BIC Value and Log Likelihood of various HMM models vs the Number of States of the Models. We can see a positive relationship between Log Likelihood and an increased number of states. Conversely, for BIC values, we see a negative relationship. Since we were unable to have any models with states > 14 converge, we conclude that the ideal hmm model for the given time window is an HMM model with 14 states.

Conclusions

Of the models that did converge, we found that an increase in states led to lower BIC values, as well as higher log likelihoods. We are aware that this trend would not continue indefinitely, and that with more states, we would eventually be overfitting the model (i.e., fitting it to the training points themselves); however, at this juncture it seems as though more states may indeed still lead to a better model, which was somewhat surprising given our prediction. Given that we were unable to have models with > 14 states converge, we settle on the best model given our constraints, a 14 state Hidden Markov Model with the following:

```
Initial state probabilities model
  pr1   pr2   pr3   pr4   pr5   pr6   pr7   pr8   pr9  pr10  pr11  pr12  pr13  pr14
0.039 0.000 0.020 0.118 0.000 0.000 0.019 0.414 0.019 0.019 0.019 0.077 0.000 0.257

Transition matrix
        toS1  toS2  toS3  toS4  toS5  toS6  toS7  toS8  toS9 toS10 toS11 toS12 toS13 toS14
fromS1  0.860 0.001 0.007 0.001 0.008 0.015 0.000 0.000 0.000 0.000 0.060 0.000 0.047 0.000
fromS2  0.000 0.793 0.046 0.007 0.025 0.000 0.062 0.000 0.000 0.006 0.011 0.019 0.026 0.005
fromS3  0.027 0.024 0.736 0.015 0.008 0.014 0.012 0.014 0.000 0.026 0.036 0.033 0.012 0.044
fromS4  0.000 0.000 0.009 0.845 0.004 0.001 0.007 0.078 0.000 0.001 0.000 0.046 0.000 0.009
fromS5  0.005 0.002 0.008 0.000 0.895 0.020 0.000 0.000 0.044 0.000 0.002 0.000 0.024 0.000
fromS6  0.023 0.000 0.002 0.001 0.014 0.837 0.000 0.000 0.051 0.006 0.016 0.000 0.049 0.000
fromS7  0.000 0.042 0.006 0.011 0.009 0.000 0.785 0.003 0.000 0.008 0.004 0.132 0.000 0.000
fromS8  0.000 0.000 0.007 0.051 0.000 0.000 0.000 0.891 0.000 0.000 0.000 0.013 0.000 0.038
fromS9  0.005 0.000 0.000 0.000 0.050 0.057 0.000 0.000 0.871 0.000 0.000 0.000 0.017 0.000
fromS10 0.018 0.013 0.016 0.003 0.012 0.008 0.000 0.009 0.000 0.847 0.043 0.010 0.014 0.006
fromS11 0.041 0.004 0.002 0.000 0.004 0.012 0.000 0.000 0.003 0.025 0.895 0.000 0.014 0.000
fromS12 0.001 0.012 0.010 0.098 0.001 0.000 0.043 0.012 0.000 0.004 0.000 0.809 0.005 0.006
fromS13 0.059 0.000 0.001 0.001 0.017 0.055 0.004 0.000 0.021 0.004 0.023 0.000 0.813 0.000
fromS14 0.000 0.003 0.003 0.031 0.001 0.000 0.001 0.037 0.000 0.000 0.001 0.001 0.000 0.921

Response parameters
Resp 1 : gaussian
     Re1.(Intercept) Re1.sd
St1            5.957  0.194
St2            2.757  0.175
St3            3.688  0.405
St4            1.448  0.098
St5           15.038  3.670
St6            8.379  0.557
St7            2.292  0.123
St8            1.078  0.105
St9           10.373  0.853
St10           4.755  0.285
St11           5.412  0.179
St12           1.864  0.114
St13           6.723  0.372
St14           0.663  0.254

Convergence info: Log likelihood converged to within tol. (relative change)
'log Lik.' -6840.719 (df=223)
AIC:  14127.44
BIC:  15784.75
```

Final Thoughts

Given the complexity of our 14-state model, we suspect that a smaller time window would have been preferable. Intuitively, modelling a 240-minute time window is likely to require more states than a 120 or 180-minute time window. If the optimized model we were seeking had < 10 states, we would have had an easier time training the models (getting more models to converge), and thus we may have had both more to choose from, as well as an optimal number of states (or an optimal range of states). But, given our time window, we feel as though we did the best job possible as can be inferred from the BIC values and log-likelihoods of the models we found to converge.