Gaussian Processes for Time Series

Bayesian Time Series Analysis

Given some time series data $D = \{(x_i, y_i)\}$ where x_i is say a time point and y_i is some reading at this time, we consider it as a regression problem:

$$y_i(\boldsymbol{x}) = f_i(\boldsymbol{x}) + \eta,$$

where f is unknown and and η is a random additive noise process, with the goal in mind to evaluate a form of f and infer a probability distribution for y given an input x. Throughout we will use the notation $\boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_n(\boldsymbol{x})]$.

Gaussian Processes

To conducdct this analysis we consider a Gaussian Process (a good intro to which lies in (Rasmussen 2003)), which is completely defined by the mean function and covariance function:

$$m(\boldsymbol{x}) = \mathbb{E}(\boldsymbol{f}(\boldsymbol{x}))$$
$$k(\boldsymbol{x}, \boldsymbol{x'}) = \mathbb{E}[\{\boldsymbol{f}(\boldsymbol{x}) - m(\boldsymbol{x})\}\{\boldsymbol{f}(\boldsymbol{x'}) - m(\boldsymbol{x'})\}]$$

and we write the process as: $f(\boldsymbol{x}) \sim \mathcal{GP}(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x'}))$, and for simplicity often set the mean function to zero.

Assumptions

A Gaussian process is defined as a collection of random variables where $\boldsymbol{x}$ is the input of time series indices, and in our case $\boldsymbol{f}(\boldsymbol{x})$ represents a time series. Hence we are assuming that we fullfil a consistency requriement, that is:

$$(y_1, y_2) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \implies y_1 \sim \mathcal{N}(\boldsymbol{\mu_1}, \boldsymbol{\Sigma}_1),$$

for the relevant submatrix $\boldsymbol{\Sigma_1}$. That is, the examination of a large set of variables does not effect the distribution of each of its subsets, a criterion automatically fulfilled when we specify a kernel function for the covariance matrix.

The model

A Gaussian process can be obtained from a Bayesian liner regression model $f(\boldsymbol{x}) = \phi(\boldsymbol{x})^\top \boldsymbol{w}$ where $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma_p})$ - then the means and kernel functions become:

$$m(\boldsymbol{x}) = \mathbb{E}[\boldsymbol{f}(\boldsymbol{x})] = \phi(\boldsymbol{x})^\top \mathbb{E}[\boldsymbol{w}] = 0$$

and

$$k(\boldsymbol{x}, \boldsymbol{x'}) = \mathbb{E}[\boldsymbol{f}(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x'})] = \phi(\boldsymbol{x})^\top \Sigma_p \phi(\boldsymbol{x'})$$

For our time series data we will use the most widely used kernel function used for this class of problem (Roberts et al. 2013), the squared exponential function:

$$k(x_i, x_j) = \sigma^2 \exp\left[-\frac{1}{2l}(x_i - x_j)^2 \right]$$

With hyperparameters σ and l, choices of which can result in very different curves, σ effectively controls the gain of the function, and l can be thought of as a smoothing parameter. It is known, (Rasmussen 2003), that this corresponds to a regression with infinite basis functions.

Fitting

Prediction using Noisy Observations

It makes sense, for time series data, to assume input time indeces $x_i = i, i = 1, \ldots$ are noiseless, and that the randomness comes with the values at each timestamp y_i. Let X be the input set for our known values, and X_* be those of the values we wish to predict with a function $\hat{\boldsymbol{f}}$ .So given a dataset we have access to noisy values $y = f(\boldsymbol{x}) + \varepsilon$ where $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$, hence the prior induced is:

$$cov(\boldsymbol{y}) = K(X, X) + \sigma_n^2 \boldsymbol{I},$$

and we can now, using consistency, write down the joint distribution of observed values $\boldsymbol{y}$ and function values $\boldsymbol{f}_*$:

$$\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{f}_* \end{bmatrix} = \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} K(X, X) + \sigma_n^2 \boldsymbol{I} & K(X, X_*) \\ K(X_*, X) & K(X_* X_*) \end{bmatrix} \right)$$

Using results on conditional Gaussian distributions (Bishop 2006), we can write down predictive equations:

$$\boldsymbol{f}_* \mid X, \boldsymbol{y}, X_* \sim \mathcal{N}\left(\hat{\boldsymbol{f}}_*, cov(\boldsymbol{f}_*) \right),$$

where

$$\hat{\boldsymbol{f}}_* = K(X_*, X)[K(X, X) + \sigma_n^2 \boldsymbol{I}]^{-1} \boldsymbol{y},$$
$$cov(\boldsymbol{f}_*) = K(X_*, X_*) - K(X_*, X)[K(X, X) + \sigma_n^2 \boldsymbol{I}]^{-1} K(X, X_*).$$

The next section will be dedicated to exploring this scheme using $\hat{\boldsymbol{f}}_*$, the conditional mean, as our predictive function on some real data - the `Fit_GP` function in our package gives a method of how to fit a Gaussian Process to time series data.

Real Data

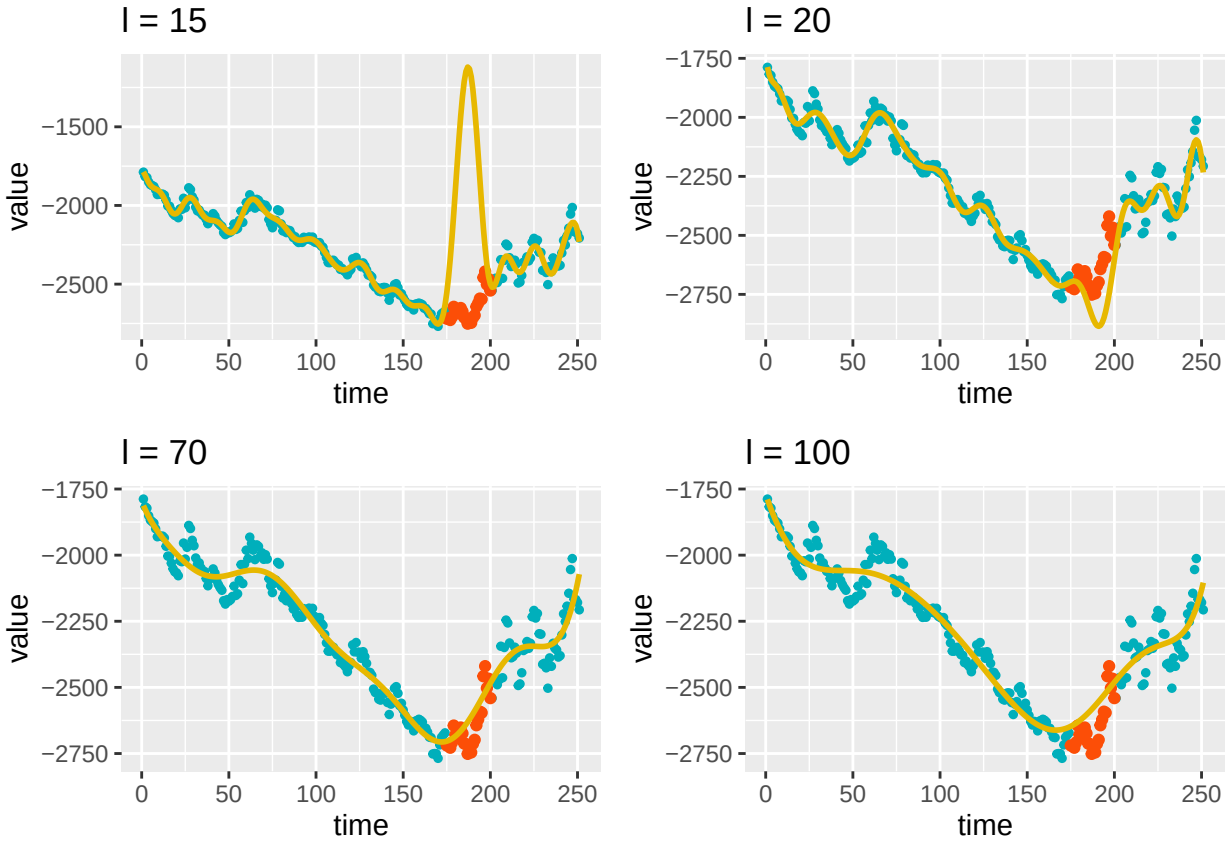
Casting our minds back to our factor analysis, we now present fits of a Gaussian process to a reduction of the undifferenced SP500 multivariate time series to a single variate factor of 251 values through time, fitted using the `myPCA` function in the `portopt` package. Fitting a Gaussian Process to the factor series corresponding to the greatest varience of the original data will allow us to analyse underlying behaviour of the SP500 data as a whole. We will now see how Gaussian processes perform when trying to interpolate and predict missing data.

Choice of Kernel parameter l for Predicting Missing Data

The problem set up here is very simple, suppose we are in the setting that a section of our factor series data has gone missing, either through collection, computer, or human error and we wish to try and learn the behaviour of our time series in order to interpolate to fill the gap in the readings. Recall our kernel function:

$$k(x_i, x_j) = \sigma^2 \exp\left[-\frac{1}{2l}(x_i - x_j)^2\right]$$

To simplify the problem let's fix our the gain parameter to $\sigma = 200$, and let's learn l to perform an interpolation for a missing set of 25 consecutive readings. Let's look at how our model prediction function, fit using the `Fit_GP` function in the `portop` package , looks for various choices of l. Here the Gaussian Process has been fit on the factor series readings with a section of 25 readings missing from $175, \ldots, 200$ for which our conditional mean serves as our prediction function, we plot it against the values the process is fitted on in blues, and the missing values in red:



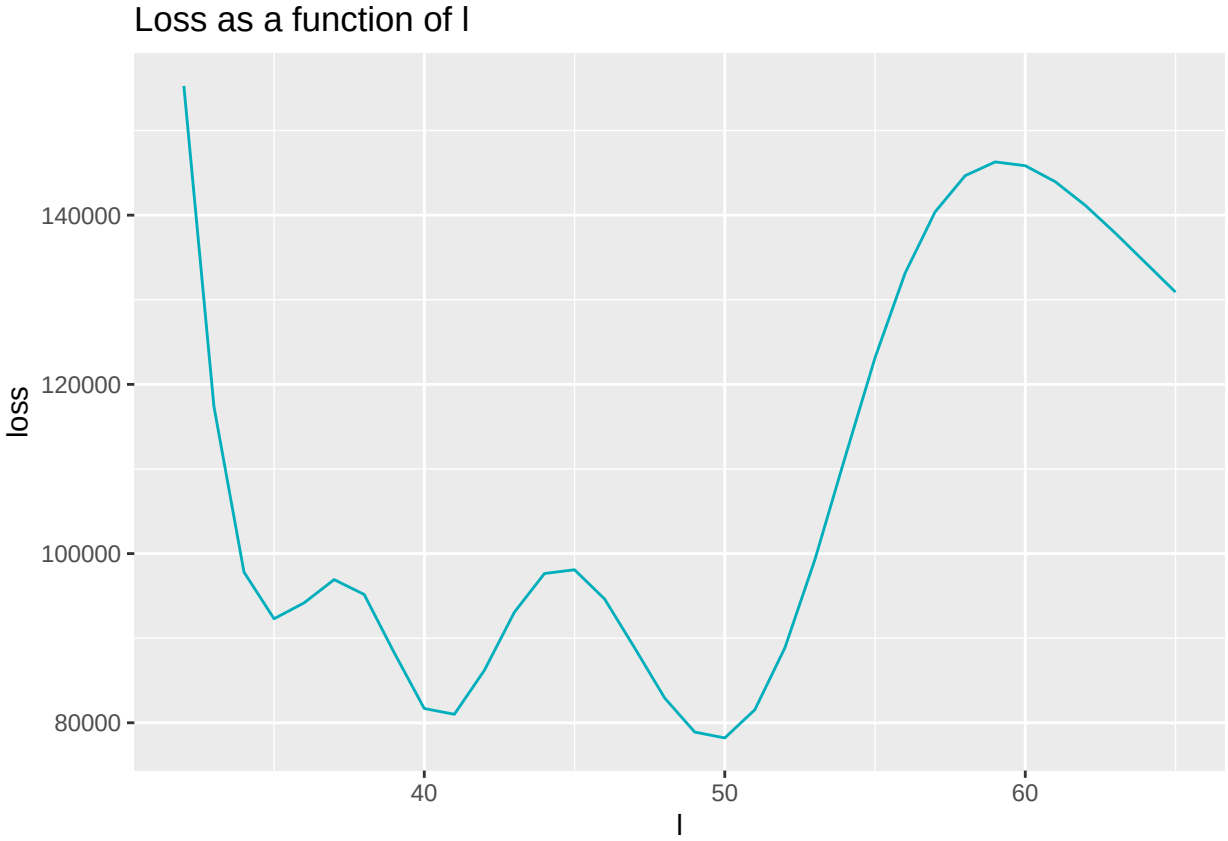
We see that increasing l decreases the flexibility of the prediction function. An interpretation of l given in (Rasmussen 2003) is that it can be thought of as a 'decay time' relative to σ, that is, how quickly should expect the process to change between readings - this interpretation is backed up by these observations as for lower values of l we see the prediction function moving further from the known values.

Let's define a loss function for our choice of l, it is defined only over the missing data indexed $j = 1, \ldots, 25$, we take the squared loss of the predictions:

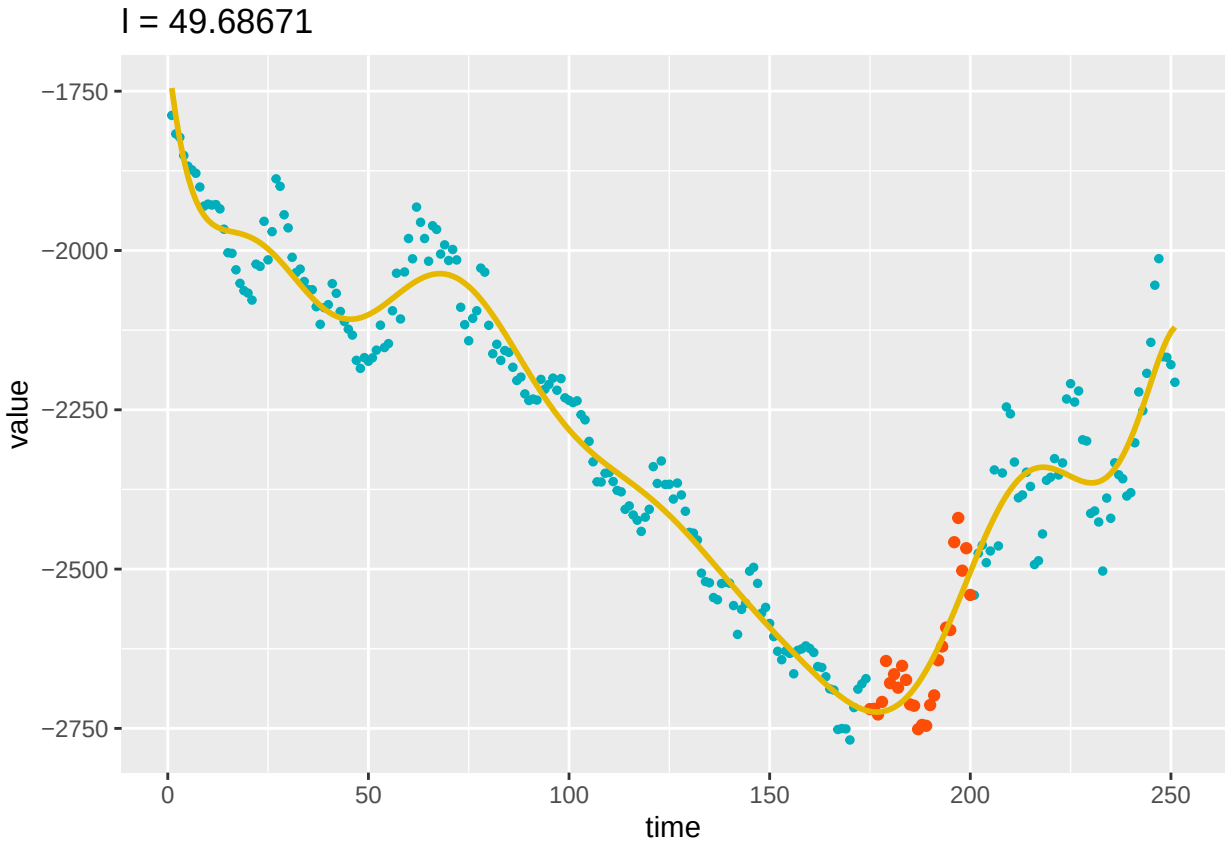
$$L(l) = \sum_{j=1}^{25} (y_{true}^{(j)} - f_j(\boldsymbol{x}))^2,$$

where $f_i(\boldsymbol{x})$ is our predictive mean at timestamp i. Let's plot this over reasonable values of l - preliminary analysis shows a loss which grows for extreme values of l so plotting over $l \in [30, 70]$ will include the relevant optimal points:

```
## [1] 2437969344
```



We see that our loss has a multiple local optima, a phenomenon frequently observed with optimisation of kernel paramenters (Rasmussen 2003), it appears we have a global minimum. Running an optimisation with the 'Brent' method indeed tells us that our squared loss is minimised for $l = 49.68671$ attaining a value of 2437969344:



Testing

So we have learned a value of l over the training set $i = 175, \ldots, 200$, let's see if our learned kernel generalises to fit unseen gaps:

Here we see our value for l tested on various sets of missing data with loss values 1778024179, 2917764245, 194541423, and 6415421161, respectively - Suggesting that the method of learning l does not generalise very well.

Conclusions

We have seen how taking a Bayesian view of modelling a time series allows us to model it as a Gaussian Process equivalent to using infinite basis functions in a regression setting, modelling as such leads to a rich arsenal for inference, notably a predictive conditional mean, variance and likelihood obtained using results from the rules of conditional Gaussian distributions. Whilst the attempt at a method to model and predict missing data values has proved to generalise weakly, an insight was gained as to how the choice of l for the squared exponential kernel effects the fit of the Gaussian Process. Further methods to explore for such problems could include utilising the predictive variance and conditional likelihood into the optimisation of of the parameters of the kernel function such as those suggested in (Roberts et al. 2013).

References

Bishop, Christopher M. 2006. *Pattern Recognition and Machine Learning*. springer.

Rasmussen, Carl Edward. 2003. "Gaussian Processes in Machine Learning." In *Summer School on Machine Learning*, 63–71. Springer.

Roberts, Stephen, Michael Osborne, Mark Ebden, Steven Reece, Neale Gibson, and Suzanne Aigrain. 2013. "Gaussian Processes for Time-Series Modelling." *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 371 (1984). The Royal Society Publishing: 20110550.