Chapter 1

Automatic Model Description

> "Not a wasted word. This has been a main point to my literary thinking all my life."
>
> – *Hunter S. Thompson*

The previous chapter showed how to automatically build structured models through a language of kernels. It also showed how to decompose the resulting models into the different types of structure present, and how to visually illustrate the type of structure captured by each component. This chapter shows how automatically describe the resulting model structures with English text.

The main idea is to treat every component of a product kernel as an adjective, or as a short phrase which modifies the description of a kernel. To see how this could work, recall that the model decomposition plots of section 1.5 showed that most of the structure in each component was determined by that component's kernel. Even across different datasets, the meaning of individual parts of kernels is consistent in some ways. For example, Per indicates repeating structure, and SE indicates smooth change over time.

This chapter also presents a system which combines automatically generated text and plots to generate reports which highlight interpretable features discovered in a variety of data sets. An example of a complete automatically-generated report can be found in appendix **??**.

The work appearing in this chapter was written in collaboration with James Robert Lloyd, Roger Grosse, Joshua B. Tenenbaum, and Zoubin Ghahramani, and was published in Lloyd et al. (2014). The procedure translating kernels into adjectives grew out of discussions between James and myself. James Lloyd wrote the code to automatically generate reports, and ran all of the experiments.

1.1 Generating descriptions of composite kernels

There are two main features of our language of GP models that allow description to be performed automatically. First, the sometimes complicated kernel expressions found by model search can always be simplified into a sum of products. As discussed in section 1.5, a sum of kernels corresponds to a sum of functions, so each product can be described separately. Second, each kernel in a product modifies the resulting model in a consistent way. Therefore, one can describe a product of kernels by concatenating descriptions of the effect of each part of the product. One part needs to be described using a noun.

For example, we can describe the product of kernels Per$\times$SE by representing Per by a noun ("a periodic function") modified by a phrase representing the effect of the SE kernel ("whose shape varies smoothly over time"). To simplify the system, we restrict the base kernels to the set {C, Lin, WN, SE, Per, and $\boldsymbol{\sigma}$}. Recall that the sigmoidal kernel $\boldsymbol{\sigma}(x, x') = \sigma(x)\sigma(x')$ allows changepoints and change-windows.

1.1.1 Simplification rules

In order to be able to use the same phrase to describe the effect of each base kernel in different circumstances, we must convert each kernel expression into a standard, simplified form.

First, we distribute all products of sums into sums of products. Then, we apply several simplification rules to the kernel expression:

- Products of two or more SE kernels can be equivalently represented by a single SE with different parameters.
- Multiplying the white-noise kernel (WN) by any stationary kernel (C, WN, SE, or Per) gives another WN kernel.
- Multiplying any kernel by the constant kernel (C) only changes the parameters of the original kernel, and so can be factored out of any product in which it appears.

After applying these rules, any composite kernel can be written as a sum of terms of the form:

$$K \prod_m \mathrm{Lin}^{(m)} \prod_n \boldsymbol{\sigma}^{(n)}, \tag{1.1}$$

where K is one of $\{\mathrm{WN}, \mathrm{C}, \mathrm{SE}, \prod_k \mathrm{Per}^{(k)}\}$ or $\{\mathrm{SE} \prod_k \mathrm{Per}^{(k)}\}$, and $\prod_i k^{(i)}$ denotes a product of kernels, each with different parameters. Superscripts denote different instances of the same kernel appearing in a product: $\mathrm{SE}^{(1)}$ can have different kernel parameters than $\mathrm{SE}^{(2)}$.

1.1.2 Describing each part of a product of kernels

Loosely speaking, each kernel in a product modifies the resulting GP model in a consistent way. This allows us to describe the contribution of each kernel in a product as an adjective, or more generally as a post-modifier of a noun. We now describe how each of the kernels in our grammar modifies a GP model:

- **Multiplication by SE** removes long range correlations from a model, since $\mathrm{SE}(x, x')$ decreases monotonically to 0 as $|x - x'|$ increases. This will convert any global correlation structure into local correlation only.

- **Multiplication by Lin** is equivalent to multiplying the function being modeled by a linear function. If $f(x) \sim \mathrm{GP}(0, k)$, then $xf(x) \sim \mathrm{GP}\left(0, \mathrm{Lin} \times k\right)$. This causes the standard deviation of the model to vary linearly, without affecting the correlation between function values.

- **Multiplication by $\boldsymbol{\sigma}$** is equivalent to multiplying the function being modeled by a sigmoid, which means that the function goes to zero before or after some point.

- **Multiplication by Per** removes correlations between all pairs of function values not close to one period apart.

- **Multiplication by any kernel** modifies the covariance in the same way as multiplying by a function drawn from a corresponding GP prior. This follows from the fact that if $f_1(x) \sim \mathrm{GP}(0, k_1)$ and $f_2(x) \sim \mathrm{GP}(0, k_2)$ then

$$\mathrm{Cov}\left[f_1(x)f_2(x), f_1(x')f_2(x')\right] = k_1(x, x')k_2(x, x'). \tag{1.2}$$

  Put more plainly, a GP whose covariance is a product of kernels has the same covariance as a product of two functions, each drawn from the corresponding GP prior. However, the distribution of $f_1 f_2$ is not GP distributed – it has higher central moments as well. This identity can be used to generate a cumbersome "worst-case" description in cases where a more concise description of the effect of a kernel is not known. For example, it is used in our system to describe the product of more than one periodic kernel.

Table 1.1 gives the corresponding description of the effect of each type of kernel in a product, written as a post-modifier.

Kernel	Postmodifier phrase
SE	whose shape changes smoothly
Per	modulated by a periodic function
Lin	with linearly varying amplitude
$\prod_k \mathrm{Lin}^{(k)}$	with polynomially varying amplitude
$\prod_k \boldsymbol{\sigma}^{(k)}$	which applies until / from [changepoint]

Table 1.1 Descriptions of the effect of each kernel, written as a post-modifier.

Table 1.2 gives the corresponding description of each kernel before it has been multiplied by any other, written as a noun phrase.

Kernel	Noun phrase
WN	uncorrelated noise
C	constant
SE	smooth function
Per	periodic function
Lin	linear function
$\prod_k \mathrm{Lin}^{(k)}$	polynomial

Table 1.2 Noun phrase descriptions of each type of kernel

1.1.3 Combining descriptions into noun phrases

In order to build a noun phrase describing a product of kernels, we choose one kernel to act as a head noun which is modified by appending descriptions of the other kernels in the product.

As an example, a kernel of the form $\mathrm{Per} \times \mathrm{Lin} \times \boldsymbol{\sigma}$ could be described as a

$$\underbrace{\mathrm{Per}}_{\text{periodic function}} \times \underbrace{\mathrm{Lin}}_{\text{with linearly varying amplitude}} \times \underbrace{\boldsymbol{\sigma}}_{\text{which applies until 1700.}}$$

where Per was chosen to be the head noun.

In our system, the head noun is chosen according to the following ordering:

$$\text{Per}, \text{WN}, \text{SE}, \text{C}, \prod_m \text{Lin}^{(m)}, \prod_n \boldsymbol{\sigma}^{(n)} \tag{1.3}$$

Combining tables 1.1 and 1.2 with ordering 1.3 provides a general method to produce descriptions of kernels.

Extensions

In practice, we also incorporate a number of other rules which help to make the descriptions shorter, easier to parse, or clearer:

- We add extra adjectives depending on kernel parameters. For example, an SE with a relatively short lengthscale might be described as "a rapidly-varying smooth function" as opposed to just "a smooth function".
- Descriptions can include kernel parameters. For example, we might write that a function is "repeating with a period of 7 days".
- Descriptions can include extra information about the model not contained in the kernel. For example, based on the slope of the posterior mean, we might "a linearly increasing function" as opposed to "a linear function".
- Some kernels can be described through pre-modifiers. For example, we might write "an approximately periodic function" as opposed to "a periodic function whose shape changes smoothly".

Ordering additive components

The reports generated by our system attempt to present the most interesting or important features of a data set first. As a heuristic, we order components by always adding next the component which most reduces the 10-fold cross-validated mean absolute error.

1.1.4 Worked example

This section shows an example of the description procedure, starting with a compound kernel containing every type of base kernel in our set:

$$\text{SE} \times (\text{WN} \times \text{Lin} + \text{CP}(\text{C}, \text{Per})). \tag{1.4}$$

This is converted to a sum of products, and the changepoint is converted into sigmoidal kernels (recall the definition of changepoint kernels in ??:

$$\mathrm{SE} \times \mathrm{WN} \times \mathrm{Lin} \; + \; \mathrm{SE} \times \mathrm{C} \times \boldsymbol{\sigma} \; + \; \mathrm{SE} \times \mathrm{Per} \times \bar{\boldsymbol{\sigma}}. \tag{1.5}$$

which is simplified using the rules in section 1.1.1 to

$$\mathrm{WN} \times \mathrm{Lin} \; + \; \mathrm{SE} \times \boldsymbol{\sigma} \; + \; \mathrm{SE} \times \mathrm{Per} \times \bar{\boldsymbol{\sigma}}. \tag{1.6}$$

To describe the first component, ($\mathrm{WN} \times \mathrm{Lin}$), the head noun description for WN, “uncorrelated noise”, is concatenated with a modifier for Lin, “with linearly increasing standard deviation”.

The second component, ($\mathrm{SE} \times \boldsymbol{\sigma}$), is described as “A smooth function with a lengthscale of [lengthscale] [units]”, corresponding to the SE, “which applies until [changepoint]”.

Finally, the third component, $\mathrm{SE}{\times}\mathrm{Per}{\times}\bar{\boldsymbol{\sigma}}$, is described as “An approximately periodic function with a period of [period] [units] which applies from [changepoint]”.

1.2 Example descriptions

In this section, we demonstrate the ability of our procedure, ABCD, to write intelligible desciptions of the structure present in two time series. The model descriptions were generated based off of models produced by the automatic search method presented in section 1.5.

1.2.1 Summarizing 400 years of solar activity

First, we show excerpts from the report automatically generated on annual solar irradiation data from 1610 to 2011. This dataset is shown in figure 1.1.

This time series has two pertinent features: First, a roughly 11-year cycle of solar activity. Second, a period lasting from 1645 to 1715 having almost no variance. This flat region corresponds to the Maunder minimum, a period in which sunspots were extremely rare (Lean et al., 1995). The Maunder minimum is an example of the type of structure which can be captured by changewindows.

The first section of each report generated by ABCDis a summary of the structure found in the dataset. Figure 1.2 show natural-language summaries of the top four com-

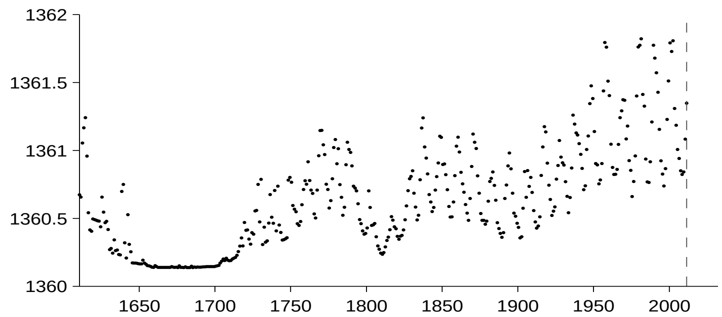

Figure 1.1 Solar irradiance data.

- A constant.
- A constant. This function applies from 1643 until 1716.
- A smooth function. This function applies until 1643 and from 1716 onwards.
- An approximately periodic function with a period of 10.8 years. This function applies until 1643 and from 1716 onwards.

Figure 1.2 Automatically generated descriptions of the first four components discovered by ABCD on the solar irradiance data set. The dataset has been decomposed into diverse structures with simple descriptions.

ponents discovered by ABCD on the solar dataset. From these short summaries, we can see that the system has identified the Maunder minimum (second component) and the 11-year solar cycle (fourth component). These components are visualized and described in figures 1.3 and 1.5, respectively. The third component, visualized in figure 1.4, captures the smooth variation over time of the overall level of solar activity.

This component is constant. This component applies from 1643 until 1716.

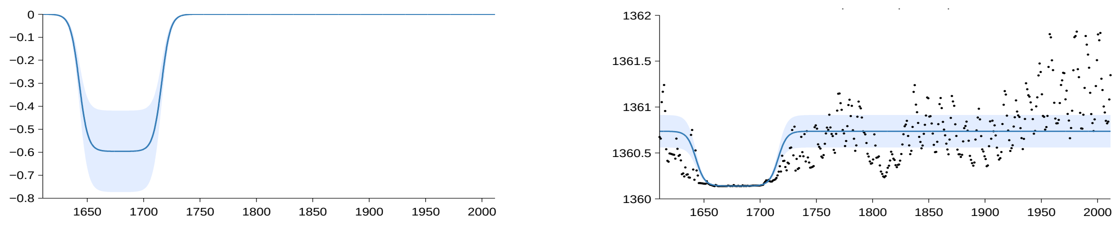

Figure 4: Pointwise posterior of component 2 (left) and the posterior of the cumulative sum of components with data (right)

Figure 1.3 Extract from an automatically-generated report describing the model component corresponding to the Maunder minimum.

This component is a smooth function with a typical lengthscale of 23.1 years. This component applies until 1643 and from 1716 onwards.

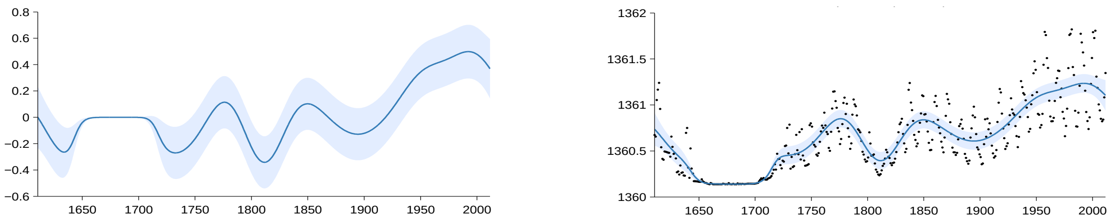

Figure 6: Pointwise posterior of component 3 (left) and the posterior of the cumulative sum of components with data (right)

Figure 1.4 Characterizing the medium-term smoothness of solar activity levels. By allowing other components to explain the periodicity, noise, and the Maunder minimum, ABCD can isolate the part of the signal best explained by a slowly-varying trend.

This component is approximately periodic with a period of 10.8 years. Across periods the shape of this function varies smoothly with a typical lengthscale of 36.9 years. The shape of this function within each period is very smooth and resembles a sinusoid. This component applies until 1643 and from 1716 onwards.

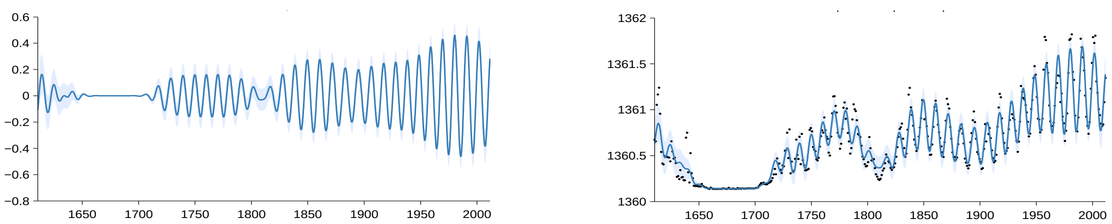

Figure 8: Pointwise posterior of component 4 (left) and the posterior of the cumulative sum of components with data (right)

Figure 1.5 This part of the report isolates and describes the approximately 11-year sunspot cycle, also noting its dissapearing during the Maunder minimum.

The complete report generated on this dataset can be found in **??**. This report also contains samples from the model posterior.

1.2.2 Describing changing noise levels

Next, we present excerpts of the description generated by our procedure on international airline passenger counts over time. The dataset and model described is the same as that shown in ??. High-level descriptions of the four components discovered are shown in figure 1.6.

- A linearly increasing function.
- An approximately periodic function with a period of 1.0 years and with linearly increasing amplitude.
- A smooth function.
- Uncorrelated noise with linearly increasing standard deviation.

Figure 1.6 Short descriptions of the four components of the airline model.

This component is approximately periodic with a period of 1.0 years and varying amplitude. Across periods the shape of this function varies very smoothly. The amplitude of the function increases linearly. The shape of this function within each period has a typical lengthscale of 6.0 weeks.

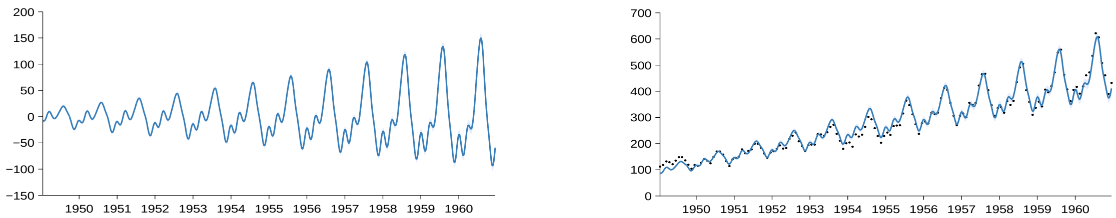

Figure 4: Pointwise posterior of component 2 (left) and the posterior of the cumulative sum of components with data (right)

Figure 1.7 Describing non-stationary periodicity in the airline data.

This component models uncorrelated noise. The standard deviation of the noise increases linearly.

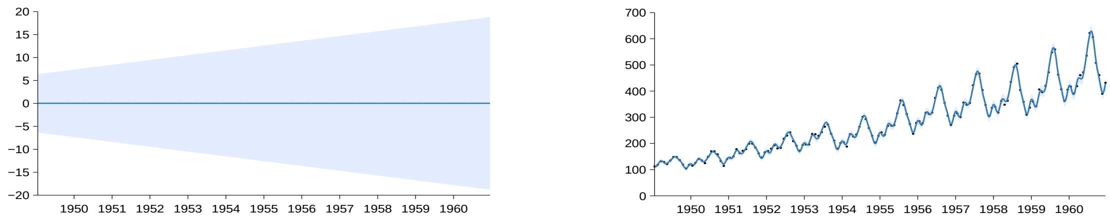

Figure 8: Pointwise posterior of component 4 (left) and the posterior of the cumulative sum of components with data (right)

Figure 1.8 Modeling time-changing variance in the airline dataset.

The second component, shown in figure 1.7, is accurately described as approximately (SE) periodic (Per) with linearly growing amplitude (Lin).

The description of the fourth component, shown in figure 1.8, expresses the fact that the scale of the unstructured noise in the model grows linearly with time.

The complete report generated on this dataset can be found in the supplementary material of Lloyd et al. (2014). Other example reports describing a wide variety of time-series can be found at `mlg.eng.cam.ac.uk/lloyd/abcdoutput/`

1.3 Related work

To the best of our knowledge, our procedure is the first example of automatic description of nonparametric statistical models. However, systems with natural language output have been built in the areas of video interpretation (Barbu et al., 2012) and automated theorem proving (Ganesalingam and Gowers, 2013).

Although not a description procedure, Durrande et al. (2013) developed an analytic method for decomposing a GP posterior into periodic and non-periodic parts, even when using non-periodic kernels.

1.4 Limitations of this approach

During development, we noted several difficulties with this approach:

- **Some kernels are hard to describe.** For instance, we did not include the RQ kernel in the text-generation procedure. This was done for several reasons. First, the RQ kernel can be equivalently expressed as a scale mixture of SE kernels. Second, it was difficult to think of a description for the hyperparameter which controls the heaviness of the tails of the RQ kernel. Third, a product of two RQ kernels does not give another RQ kernel, which raises the question of how to concisely describe products of RQ kernels.

- **Reliance on additivity.** Much of the modularity of the description procedure is due to the additive decomposition. However, additivity is lost under any nonlinear transformation of the output. Such warpings can be learned (Snelson et al., 2004), but descriptions of transformations of the data may not be as clear to the end user.

- **Difficulty of expressing uncertainty.** A natural extension to the model search procedure would be to report a posterior distribution on structures and kernel parameters, rather than point estimates. Describing uncertainty about the hyperparameters of a particular structure may be feasible, but describing even a few most-probable structures might result in excessively long reports.

Source code

Source code to perform all experiments is available at
http://www.github.com/jamesrobertlloyd/gpss-research.

1.5 Conclusions

This chapter presented a system which automatically generates detailed reports that describe patterns captured by a GP model. The properties of GPs and the kernels being used allowed a modular description, avoiding an exponential blowup in the number of special cases that needed to be considered.

Combined with the model search of section 1.5, this gives a procedure combining all the elements listed in **??**: an open-ended language of models, a search through model space, a model comparison procedure, and a model description procedure. Each particular element used in the procedure presented here is merely a proof-of-concept. However, even this simple prototype demonstrated the ability to discover and describe a variety of patterns in time series.

References

Andrei Barbu, Alexander Bridge, Zachary Burchill, Dan Coroian, Sven Dickinson, Sanja Fidler, Aaron Michaux, Sam Mussman, Siddharth Narayanaswamy, Dhaval Salvi, Lara Schmidt, Jiangnan Shangguan, Jeffrey M. Siskind, Jarrell Waggoner, Song Wang, Jinlian Wei, Yifan Yin, and Zhiqi Zhang. Video in sentences out. In *Conference on Uncertainty in Artificial Intelligence*, 2012. (page 10)

Nicolas Durrande, James Hensman, Magnus Rattray, and Neil D. Lawrence. Gaussian process models for periodicity detection. *arXiv preprint arXiv:1303.7090*, 2013. (page 10)

M. Ganesalingam and Timonthy W. Gowers. A fully automatic problem solver with human-style output. *arXiv preprint arXiv:1309.4501*, 2013. (page 10)

Judith Lean, Juerg Beer, and Raymond Bradley. Reconstruction of solar irradiance since 1610: Implications for climate change. *Geophysical Research Letters*, 22(23): 3195–3198, 1995. (page 6)

James Robert Lloyd, David Duvenaud, Roger B. Grosse, Joshua B. Tenenbaum, and Zoubin Ghahramani. Automatic construction and natural-language description of nonparametric regression models. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2014. (pages 1 and 10)

Edward Snelson, Carl E. Rasmussen, and Zoubin Ghahramani. Warped Gaussian processes. *Advances in Neural Information Processing Systems 16*, pages 337–344, 2004. (page 10)