Chapter 1

Automatic Model Description

The previous chapter showed how to automatically build structured models through a language of kernels, 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.

Most of the structure present in each component was determined by that component's kernel. Even across different datasets, the meaning of the individual parts of those kernels is relatively consistent. For example, Per usually indicates some sort of repeating structure, and SE usually indicates smooth change over time. This chapter shows how to take advantage of this modularity to develop a method for automatically describing the structure represented by components of structured GP models through 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.

Combining model search, plots, and text, we present a system which automatically generates reports which highlight interpretable features discovered in a variety of data sets. An example of a complete automatically-generated report can be found in ??.

The work appearing in this chapter was written in collaboration with James Robert Lloyd, Roger Grosse, Joshua B. Tenenbaum, Zoubin Ghahramani, and was published in (?). 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 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 the 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, to generate a description of a product of kernels, we can describe one kernel using a noun with all others described using adjectives.

For example, we can describe the product of kernels Per×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 C, Lin, WN, SE, Per, and $\boldsymbol{\sigma}$ (allowing changepoints and change-windows).

1.1.1 Simplification Rules

In order to be able to use the same adjectives to for each type of kernel in different circumstances, we must convert each kernel expression into a standard, simplified form.

We do this by first distributing all products of sums into a sum of products. Then, we apply several simplifications 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 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 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 $\{\mathrm{SE} \prod_k \mathrm{Per}^{(k)}\}$, where $\prod_i k^{(i)}$ denotes a product of kernels, each with different parameters. We use superscripts to distinguish between 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

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, $\text{SE}(x, x')$ 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 $f(x) \sim \text{GP}(0, k)$, then $xf(x) \sim \text{GP}\left(0, \text{Lin} \times k\right)$. This causes the standard deviation of the model to vary linearly without affecting the correlation.

- **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** modifies the correlation structure in the same way as multiplying the function by an independent periodic function. If $f_1(x) \sim \text{GP}(0, k_1)$ and $f_2(x) \sim \text{GP}(0, k_2)$ then

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

  In plain english, this identity says that a GP whose kernel is a product of kernels has the same covariance (but not necessarily the same higher moments) as a product of two functions, each drawing from the corresponding GP prior. This identity holds for any two kernels, and can be used to generate a cumbersome “worst-case” description in the case where a more meaningful description of the effect of a kernel is not obvious.

Constructing a complete description of a product of kernels

We choose one kernel to act as a noun which is then described by the functions it encodes for when unmodified e.g. ‘smooth function’ for SE. Modifiers corresponding to the other kernels in the product are then appended to this description, forming a noun phrase of the form:

$$\text{Determiner} + \text{Premodifiers} + \text{Noun} + \text{Postmodifiers}$$

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

$$\underbrace{\text{SE}}_{\text{approximately}} \times \underbrace{\text{Per}}_{\text{periodic function}} \times \underbrace{\text{Lin}}_{\text{with linearly growing amplitude}} \times \underbrace{\boldsymbol{\sigma}}_{\text{until 1700.}}$$

where Per has been selected as the head noun.

In principle, any assignment of kernels in a product to these different phrasal roles is possible, but in practice we found certain assignments to produce more interpretable phrases than others. The head noun is chosen according to the following ordering:

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

i.e. Per is always chosen as the head noun when present.

Ordering additive components The reports generated by ABCD 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.

1.1.3 Worked Example

Suppose we start with a kernel of the form

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

This is converted to a sum of products:

$$\text{SE} \times \text{WN} \times \text{Lin} + \text{SE} \times \text{C} \times \boldsymbol{\sigma} + \text{SE} \times \text{Per} \times \bar{\boldsymbol{\sigma}}.$$

which is simplified to

$$\text{WN} \times \text{Lin} + \text{SE} \times \boldsymbol{\sigma} + \text{SE} \times \text{Per} \times \bar{\boldsymbol{\sigma}}.$$

To describe the first component, the head noun description for WN, 'uncorrelated noise', is concatenated with a modifier for Lin, 'with linearly increasing standard deviation'. The second component is described as 'A smooth function with a lengthscale of [lengthscale] [units]', corresponding to the SE, 'which applies until [changepoint]', which corresponds to the $\boldsymbol{\sigma}$. Finally, the third component is described as 'An approximately periodic function with a period of [period] [units] which applies from [changepoint]'.

We demonstrate the ability of our procedure to discover and describe a variety of patterns on two time series.

1.2 Summarizing 400 Years of Solar Activity

We show excerpts from the report automatically generated on annual solar irradiation data from 1610 to 2011 (fig. 1.1). This time series has two pertinent features: a roughly 11-year cycle of solar activity, and a period lasting from 1645 to 1715 with much smaller variance than the rest of the dataset. This flat region corresponds to the Maunder minimum, a period in which sunspots were extremely rare (?). ABCD clearly identifies these two features, as discussed below.

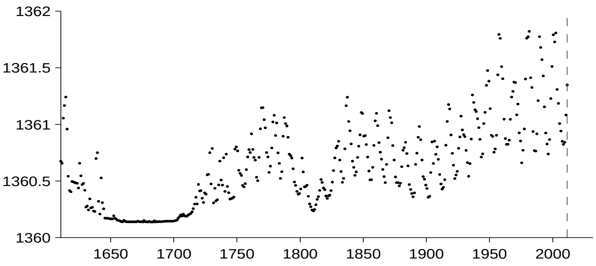
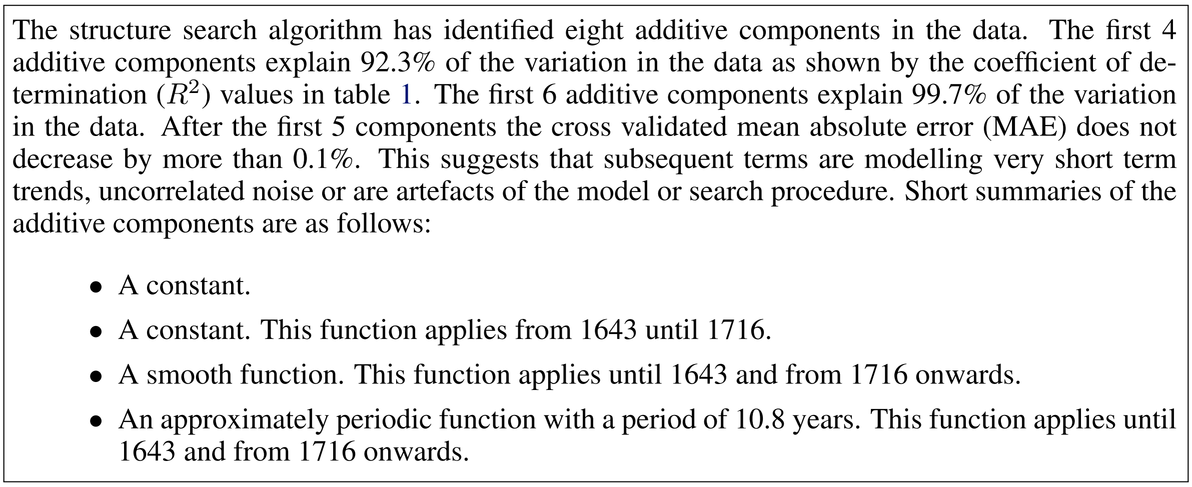


Fig. 1.1 Solar irradiance data.

Figure 1.2 shows the natural-language summaries of the top four components chosen by ABCD. From these short summaries, we can see that our system has identified the Maunder minimum (second component) and 11-year solar cycle (fourth component). These components are visualized in figures 1.3 and 1.5, respectively. The third component corresponds to long-term trends, as visualized in fig. 1.4.

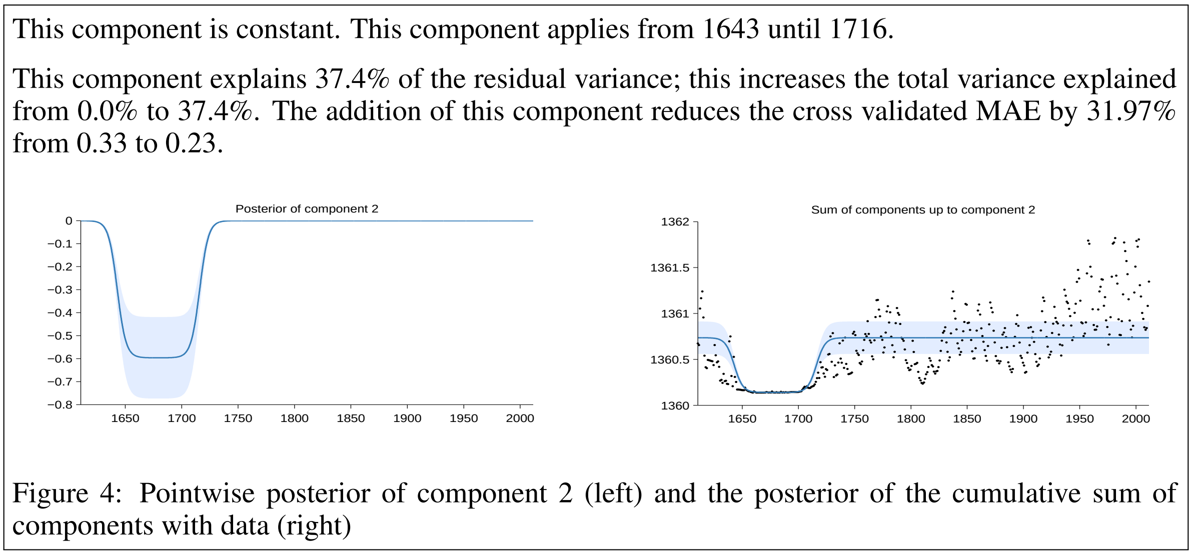
The complete report generated on this dataset can be found in ??.



The structure search algorithm has identified eight additive components in the data. The first 4 additive components explain 92.3% of the variation in the data as shown by the coefficient of determination (R^2) values in table 1. The first 6 additive components explain 99.7% of the variation in the data. After the first 5 components the cross validated mean absolute error (MAE) does not decrease by more than 0.1%. This suggests that subsequent terms are modelling very short term trends, uncorrelated noise or are artefacts of the model or search procedure. Short summaries of the additive components are as follows:

- 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.

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



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

This component explains 37.4% of the residual variance; this increases the total variance explained from 0.0% to 37.4%. The addition of this component reduces the cross validated MAE by 31.97% from 0.33 to 0.23.

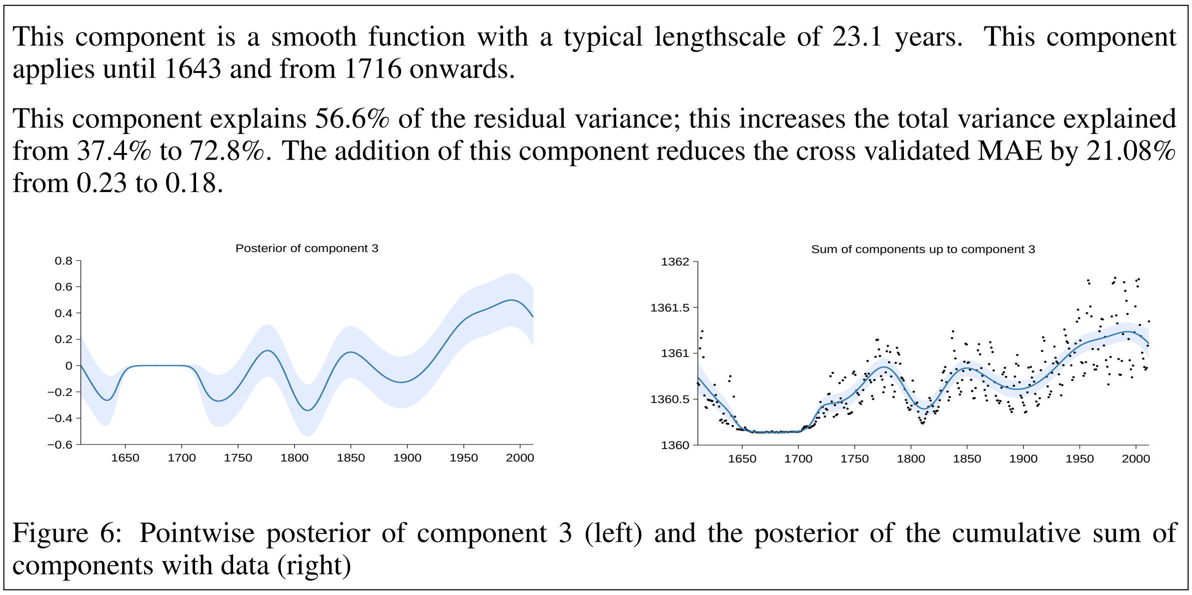
Figure 4: Pointwise posterior of component 2 (left) and the posterior of the cumulative sum of components with data (right)

Fig. 1.3 One of the learned components corresponds to the Maunder minimum.

1.3 Describing Heteroscedasticity in Air Traffic Data

Next, we present the analysis generated by our procedure on international airline passenger data (fig. 1.6). The model constructed by ABCD has four components: Lin + SE × Per × Lin + SE + WN × Lin, with descriptions given in fig. 1.7.

The second component, shown in fig. 1.8, is accurately described as approximately (SE) periodic (Per) with linearly growing amplitude (Lin). By multiplying a white noise

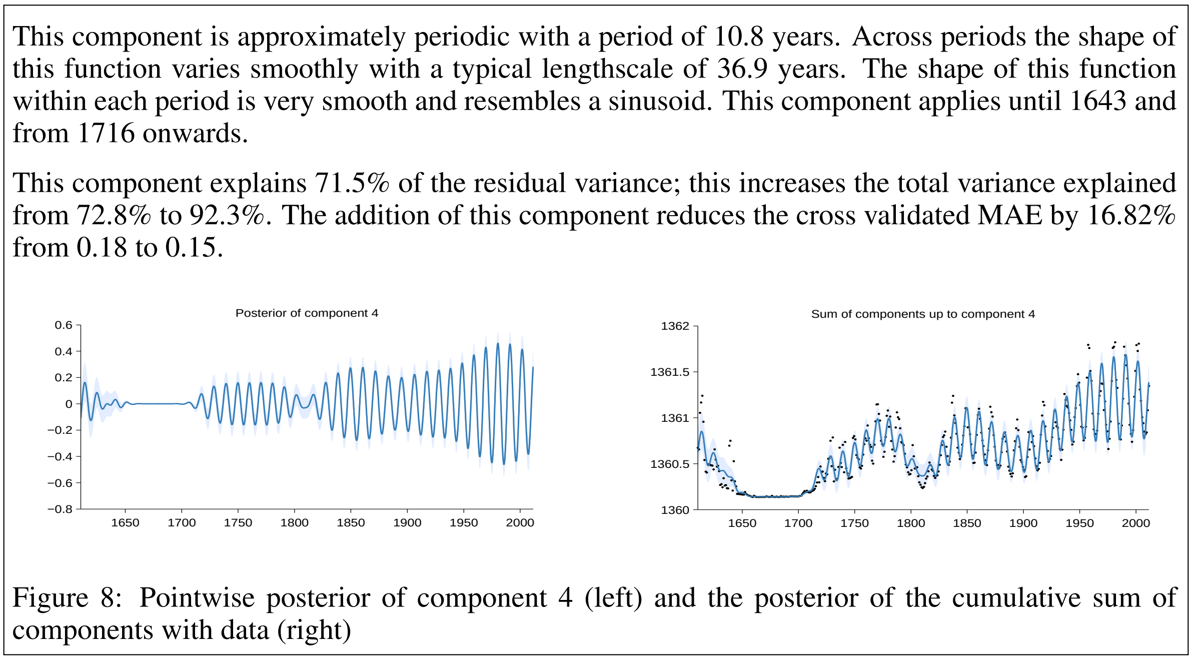


This component is a smooth function with a typical lengthscale of 23.1 years. This component applies until 1643 and from 1716 onwards.

This component explains 56.6% of the residual variance; this increases the total variance explained from 37.4% to 72.8%. The addition of this component reduces the cross validated MAE by 21.08% from 0.23 to 0.18.

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

Fig. 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.

This component explains 71.5% of the residual variance; this increases the total variance explained from 72.8% to 92.3%. The addition of this component reduces the cross validated MAE by 16.82% from 0.18 to 0.15.

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

Fig. 1.5 Extract from an automatically-generated report describing the model components discovered by automatic model search. This part of the report isolates and describes the approximately 11-year sunspot cycle, also noting its disappearance during the 16th century, a time known as the Maunder minimum (**?**).

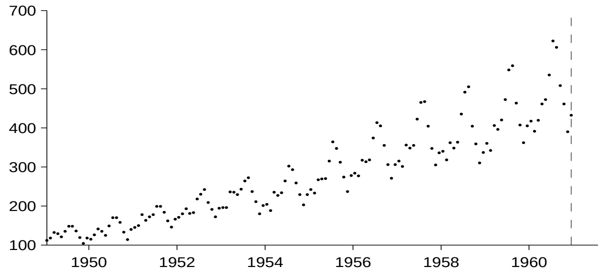


Fig. 1.6 International airline passenger monthly volume (e.g. **?**).

The structure search algorithm has identified four additive components in the data. The first 2 additive components explain 98.5% of the variation in the data as shown by the coefficient of determination (R^2) values in table 1. The first 3 additive components explain 99.8% of the variation in the data. After the first 3 components the cross validated mean absolute error (MAE) does not decrease by more than 0.1%. This suggests that subsequent terms are modelling very short term trends, uncorrelated noise or are artefacts of the model or search procedure. Short summaries of the additive components are as follows:

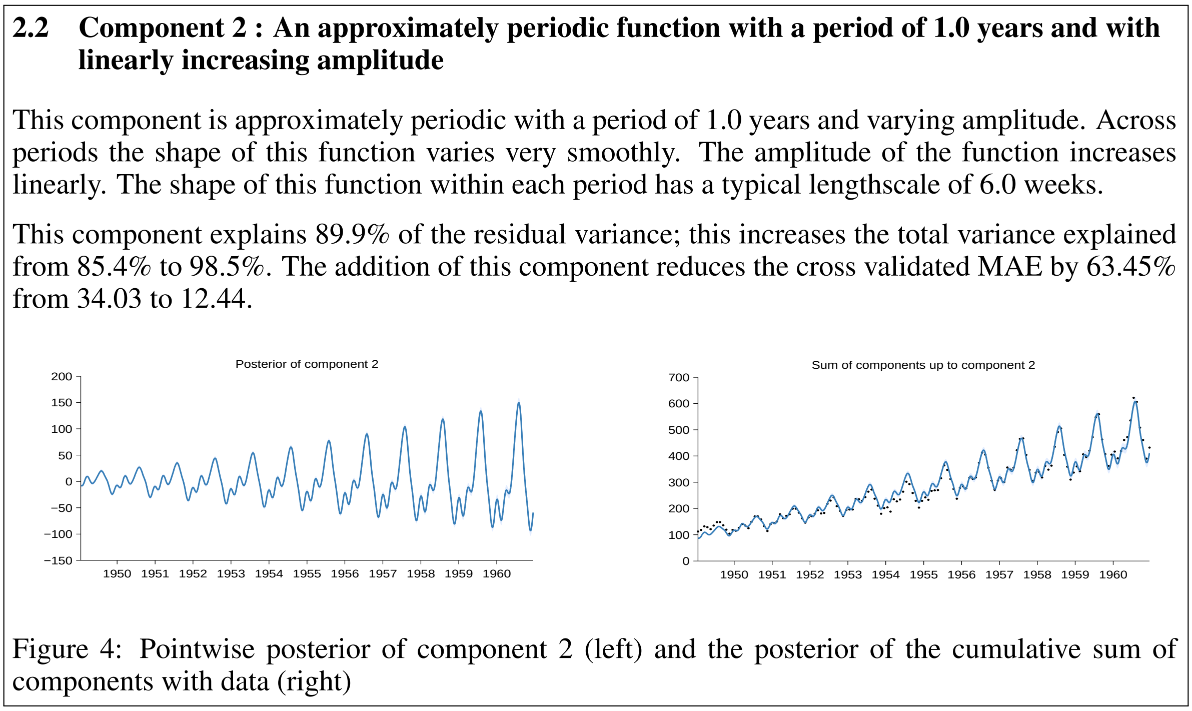
- A linearly increasing function.
- An approximately periodic function with a period of 1.0 years and with linearly increasing amplitude.
- A smooth function.
- Uncorrelated noise with linearly increasing standard deviation.

#	R^2 (%)	ΔR^2 (%)	Residual R^2 (%)	Cross validated MAE	Reduction in MAE (%)
-	-	-	-	280.30	-
1	85.4	85.4	85.4	34.03	87.9
2	98.5	13.2	89.9	12.44	63.4
3	99.8	1.3	85.1	9.10	26.8
4	100.0	0.2	100.0	9.10	0.0

Fig. 1.7 Short descriptions and summary statistics for the four components of the airline model.

kernel by a linear kernel, the model is able to express heteroscedasticity (figure 1.9).

The complete report generated on this dataset can be found in the supplementary material of (**?**). Other example reports can be found at `mlg.eng.cam.ac.uk/Lloyd/abcdoutput/`



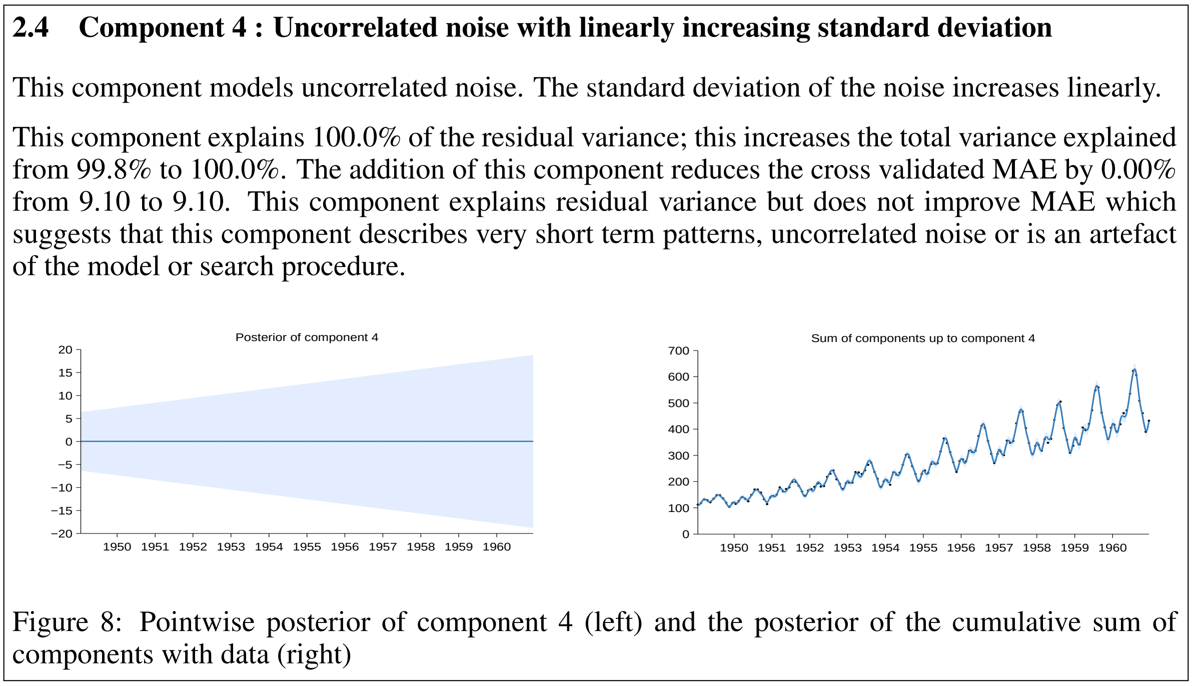
2.2 Component 2 : An approximately periodic function with a period of 1.0 years and with linearly increasing amplitude

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.

This component explains 89.9% of the residual variance; this increases the total variance explained from 85.4% to 98.5%. The addition of this component reduces the cross validated MAE by 63.45% from 34.03 to 12.44.

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

Fig. 1.8 Capturing non-stationary periodicity in the airline data



2.4 Component 4 : Uncorrelated noise with linearly increasing standard deviation

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

This component explains 100.0% of the residual variance; this increases the total variance explained from 99.8% to 100.0%. The addition of this component reduces the cross validated MAE by 0.00% from 9.10 to 9.10. This component explains residual variance but does not improve MAE which suggests that this component describes very short term patterns, uncorrelated noise or is an artefact of the model or search procedure.

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

Fig. 1.9 Modeling heteroscedasticity in the airline dataset.

1.4 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 (**?**) and automated theorem proving (**?**).

Source Code

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

1.5 Conclusion

Towards the goal of automating statistical modeling we have presented a system which constructs an appropriate model from an open-ended language and automatically generates detailed reports that describe patterns in the data captured by the model. We have demonstrated that our procedure can discover and describe a variety of patterns on several time series. We believe this procedure has the potential to make powerful statistical model-building techniques accessible to non-experts.