0.1 Kernels

0.1.1 Base kernels

For scalar-valued inputs, the white noise (WN), constant (C), linear (Lin), squared exponential (SE), and periodic kernels (Per) are defined as follows:

$$\mathrm{WN}(x, x') = \sigma^2 \delta_{x,x'} \tag{1}$$
$$\mathrm{C}(x, x') = \sigma^2 \tag{2}$$
$$\mathrm{Lin}(x, x') = \sigma^2 (x - \ell)(x' - \ell) \tag{3}$$
$$\mathrm{SE}(x, x') = \sigma^2 \exp\left(-\frac{(x-x')^2}{2\ell^2}\right) \tag{4}$$
$$\mathrm{Per}(x, x') = \sigma^2 \frac{\exp\left(\frac{\cos \frac{2\pi(x-x')}{p}}{\ell^2}\right) - I_0\left(\frac{1}{\ell^2}\right)}{\exp\left(\frac{1}{\ell^2}\right) - I_0\left(\frac{1}{\ell^2}\right)} \tag{5}$$

where $\delta_{x,x'}$ is the Kronecker delta function, I_0 is the modified Bessel function of the first kind of order zero and other symbols are parameters of the kernel functions.

0.1.2 Changepoints and changewindows

The changepoint, $\mathrm{CP}(\cdot, \cdot)$ operator is defined as follows:

$$\begin{aligned}\mathrm{CP}(k_1, k_2)(x, x') = \;& \sigma(x) k_1(x, x') \sigma(x') \\ & + (1 - \sigma(x)) k_2(x, x')(1 - \sigma(x'))\end{aligned} \tag{6}$$

where $\sigma(x) = 0.5 \times (1 + \tanh(\frac{\ell - x}{s}))$. This can also be written as

$$\mathrm{CP}(k_1, k_2) = \boldsymbol{\sigma} k_1 + \bar{\boldsymbol{\sigma}} k_2 \tag{7}$$

where $\boldsymbol{\sigma}(x, x') = \sigma(x)\sigma(x')$ and $\bar{\boldsymbol{\sigma}}(x, x') = (1 - \sigma(x))(1 - \sigma(x'))$.

Changewindow, $\mathrm{CW}(\cdot, \cdot)$, operators are defined similarly by replacing the sigmoid, $\sigma(x)$, with a product of two sigmoids.

0.1.3 Properties of the periodic kernel

A simple application of l'Hôpital's rule shows that

$$\mathrm{Per}(x, x') \to \sigma^2 \cos\left(\frac{2\pi(x - x')}{p}\right) \quad \text{as} \quad \ell \to \infty. \tag{8}$$

This limiting form is written as the cosine kernel (cos).

0.2 Model construction / search

0.2.1 Overview

The model construction phase of ABCD starts with the kernel equal to the noise kernel, WN. New kernel expressions are generated by applying search operators to the current kernel. When new base kernels are proposed by the search operators, their parameters are randomly initialised with several restarts. Parameters are then optimized by conjugate gradients to maximise the likelihood of the data conditioned on the kernel parameters. The kernels are then scored by the Bayesian information criterion and the top scoring kernel is selected as the new kernel. The search then proceeds by applying the search operators to the new kernel i.e. this is a greedy search algorithm.

In all experiments, 10 random restarts were used for parameter initialisation and the search was run to a depth of 10.

0.2.2 Search operators

ABCD is based on a search algorithm which used the following search operators

$$\mathcal{S} \to \mathcal{S} + \mathcal{B} \tag{9}$$

$$\mathcal{S} \to \mathcal{S} \times \mathcal{B} \tag{10}$$

$$\mathcal{B} \to \mathcal{B}' \tag{11}$$

where $\mathcal{S}$ represents any kernel subexpression and $\mathcal{B}$ is any base kernel within a kernel expression i.e. the search operators represent addition, multiplication and replacement.

To accommodate changepoint/window operators we introduce the following additional operators

$$\mathcal{S} \rightarrow \mathrm{CP}(\mathcal{S}, \mathcal{S}) \tag{12}$$
$$\mathcal{S} \rightarrow \mathrm{CW}(\mathcal{S}, \mathcal{S}) \tag{13}$$
$$\mathcal{S} \rightarrow \mathrm{CW}(\mathcal{S}, \mathrm{C}) \tag{14}$$
$$\mathcal{S} \rightarrow \mathrm{CW}(\mathrm{C}, \mathcal{S}) \tag{15}$$

where C is the constant kernel. The last two operators result in a kernel only applying outside or within a certain region.

Based on experience with typical paths followed by the search algorithm we introduced the following operators

$$\mathcal{S} \rightarrow \mathcal{S} \times (\mathcal{B} + \mathrm{C}) \tag{16}$$
$$\mathcal{S} \rightarrow \mathcal{B} \tag{17}$$
$$\mathcal{S} + \mathcal{S}' \rightarrow \mathcal{S} \tag{18}$$
$$\mathcal{S} \times \mathcal{S}' \rightarrow \mathcal{S} \tag{19}$$

where $\mathcal{S}'$ represents any other kernel expression. Their introduction is currently not rigorously justified.

0.3 Predictive accuracy

Interpolation To test the ability of the methods to interpolate, we randomly divided each data set into equal amounts of training data and testing data. We trained each algorithm on the training half of the data, produced predictions for the remaining half and then computed the root mean squared error (RMSE). The values of the RMSEs are then standardised by dividing by the smallest RMSE for each data set i.e. the best performance on each data set will have a value of 1.

Figure **??** shows the standardised RMSEs for the different algorithms. The box plots show that all quartiles of the distribution of standardised RMSEs are lower for both versions of ABCD. The median for ABCD-accuracy is 1; it is the best performing algorithm on 7 datasets. The largest outliers of ABCD and spectral kernels are similar in value.

Changepoints performs slightly worse than MKL despite being strictly more general than Changepoints. The introduction of changepoints allows for more structured

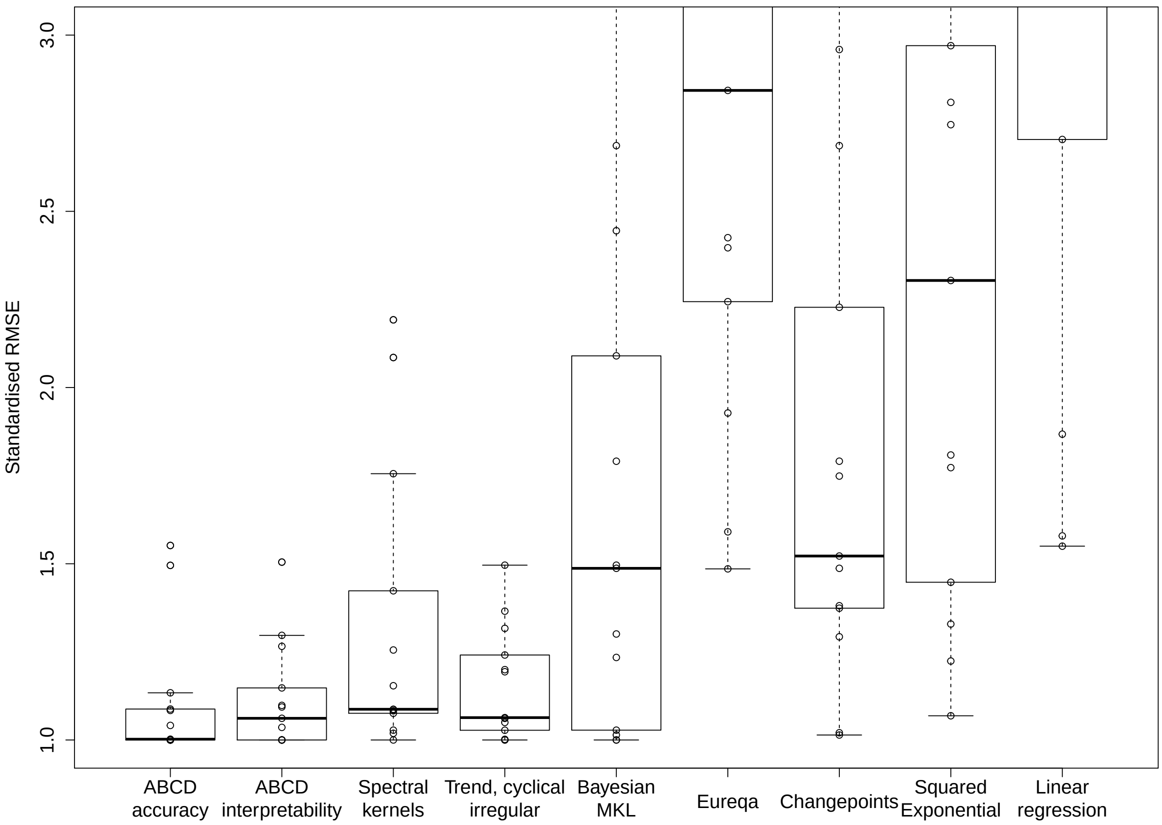

Fig. 1 Box plot of standardised RMSE (best performance = 1) on 13 interpolation tasks.

models, but it introduces parametric forms into the regression models (i.e. the sigmoids expressing the changepoints). This results in worse interpolations at the locations of the change points, suggesting that a more robust modeling language would require a more flexible class of changepoint shapes or improved inference (e.g. fully Bayesian inference over the location and shape of the changepoint).

Eureqa is not suited to this task and performs poorly. The models learned by Eureqa tend to capture only broad trends of the data since the fine details are not well explained by parametric forms.

0.3.1 Tabels of standardised RMSEs

See table **??** for raw interpolation results and table **??** for raw extrapolation results. The rows follow the order of the datasets in the rest of the supplementary material. The following abbreviations are used: ABCD-accuracy (ABCD-acc), ABCD-interpretability ((ABCD-int), Spectral kernels (SP), Trend-cyclical-irregular (TCI), Bayesian MKL (MKL), Eureqa (EL), Changepoints (CP), Squared exponential (SE) and Linear regression (Lin).

ABCD-acc	ABCD-int	SP	TCI	MKL	EL	CP	SE	Lin
1.04	1.00	2.09	1.32	3.20	5.30	3.25	4.87	5.01
1.00	1.27	1.09	1.50	1.50	3.22	1.75	2.75	3.26
1.00	1.00	1.09	1.00	2.69	26.20	2.69	7.93	10.74
1.09	1.04	1.00	1.00	1.00	1.59	1.37	1.33	1.55
1.00	1.06	1.08	1.06	1.01	1.49	1.01	1.07	1.58
1.50	1.00	2.19	1.37	2.09	7.88	2.23	6.19	7.36
1.55	1.50	1.02	1.00	1.00	2.40	1.52	1.22	6.28
1.00	1.30	1.26	1.24	1.49	2.43	1.49	2.30	3.20
1.00	1.09	1.08	1.06	1.30	2.84	1.29	2.81	3.79
1.08	1.00	1.15	1.19	1.23	42.56	1.38	1.45	2.70
1.13	1.00	1.42	1.05	2.44	3.29	2.96	2.97	3.40
1.00	1.15	1.76	1.20	1.79	1.93	1.79	1.81	1.87
1.00	1.10	1.03	1.03	1.03	2.24	1.02	1.77	9.97

Table 1 Interpolation standardised RMSEs

0.4 Guide to the automatically generated reports

Additional supplementary material to this paper is 13 reports automatically generated by ABCD. A link to these reports will be maintained at `http://mlg.eng.cam.ac.uk/lloyd/`. We recommend that you read the report for '01-airline' first and review the reports that follow afterwards more briefly. '02-solar' is discussed in the main text. '03-mauna' analyses a dataset mentioned in the related work. '04-wheat' demonstrates changepoints being used to capture heteroscedasticity. '05-temperature' extracts an exactly periodic pattern from noisy data. '07-call-centre' demonstrates a large discontinuity being modeled by a changepoint. '10-sulphuric' combines many changepoints to create a highly structured model of the data. '12-births' discovers multiple periodic components.

ABCD-acc	ABCD-int	SP	TCI	MKL	EL	CP	SE	Lin
1.14	2.10	1.00	1.44	4.73	3.24	4.80	32.21	4.94
1.00	1.26	1.21	1.03	1.00	2.64	1.03	1.61	1.07
1.40	1.00	1.32	1.29	1.74	2.54	1.74	1.85	3.19
1.07	1.18	3.00	3.00	3.00	1.31	1.00	3.03	1.02
1.00	1.00	1.03	1.00	1.35	1.28	1.35	2.72	1.51
1.00	2.03	3.38	2.14	4.09	6.26	4.17	4.13	4.93
2.98	1.00	11.04	1.80	1.80	493.30	3.54	22.63	28.76
3.10	1.88	1.00	2.31	3.13	1.41	3.13	8.46	4.31
1.00	2.05	1.61	1.52	2.90	2.73	3.14	2.85	2.64
1.00	1.45	1.43	1.80	1.61	1.97	2.25	1.08	3.52
2.16	2.03	3.57	2.23	1.71	2.23	1.66	1.89	1.00
1.06	1.00	1.54	1.56	1.85	1.93	1.84	1.66	1.96
3.03	4.00	3.63	3.12	3.16	1.00	5.83	5.35	4.25

Table 2 Extrapolation standardised RMSEs

0.5 Discussion

0.5.1 Why haven't structured kernels been built for SVMs?

Because without marginal likelihood to tell you which structure is present in your data, it's not clear how to choose which kernel to use without cross-validation.

0.6 Ingredients of an automatic statistician

? asks "How can an artificial intelligence do statistics? ... It needs not just an inference engine, but also a way to construct new models and a way to check models. Currently, those steps are performed by humans, but the AI would have to do it itself."

In this section, we discuss in more detail the elements we believe are required to build an artificial intelligence that can do statistics.

1. An open-ended language of models Many statistical procedures consider all models in a class of fixed size - for example, graphical model construction algorithms [1] search over connectivity graphs for a given set of nodes. While these methods can be powerful, human statisticians are capable of deriving novel model classes when required by the modelling task. An automatic search through an open-ended class of models can achieve some of this flexibility, growing the complexity of the model to fit the task at hand, and possibly combining existing structures in novel ways.

2. Searching through model space An open-ended space of models cannot be searched exhaustively. Just as human researchers iteratively refine their models, search procedures can propose new search directions based on the results of previous model fits. Because any search in an open-ended space must start with relatively simple models before moving on to more complex ones, any model search in an open-ended space will likely resemble a model-building procedure.

3. Model comparison and checking model fit (2) An automatic statistician should be able to question the models it has constructed, and formal procedures from model checking provide a way for it to do this. **?** review the literature on model checking. (!) In this work, we use approximate marginal likelihood to compare models, penalizing complexity using the Bayesian Information Criterion as a heuristic.

(2) JL: Two section haps - or are they pletely intertwined?

4. Describing models Part of the value of statistical models comes from enabling humans to understand a dataset or a phenomenon. Furthermore, a clear description of the statistical structure found in a dataset helps a user to notice when the dataset has errors, the wrong question was asked, the model-building procedure failed to capture known structure, a relevant piece of data or constraint is missing, or when a novel statistical structure has been found.

In this work, we demonstrate that the properties of Gaussian processes allow for a modular description generation procedure. Whether or not such modularity and interpretability is present in other open-ended model classes is an open question.