Building an Automatic Statistician

Zoubin Ghahramani

Department of Engineering
University of Cambridge

zoubin@eng.cam.ac.uk
http://learning.eng.cam.ac.uk/zoubin/

ALT-Discovery Science Conference, October 2014

James Robert Lloyd
Cambridge

David Duvenaud
Cambridge → Harvard

Roger Grosse
MIT → Toronto

Josh Tenenbaum
MIT

There is a growing need for data analysis

- We live in an era of abundant data

- The McKinsey Global Institute claim
  - *“The United States alone faces a shortage of 140,000 to 190,000 people with analytical expertise and 1.5 million managers and analysts with the skills to understand and make decisions based on the analysis of big data.”*

- Diverse fields increasingly relying on expert statisticians, machine learning researchers and data scientists e.g.
  - Computational sciences (e.g. biology, astronomy, . . .)
  - Online advertising
  - Quantitative finance
  - . . .

GOALS OF THE AUTOMATIC STATISTICIAN PROJECT

- Provide a set of tools for understanding data that require minimal expert input

- Uncover challenging research problems in e.g.
  - Automated inference
  - Model construction and comparison
  - Data visualisation and interpretation

- Advance the field of machine learning in general

Background

- Probabilistic modelling
- Model selection and marginal likelihoods
- Bayesian nonparametrics
- Gaussian processes

Probabilistic Modelling

- A model describes data that one could observe from a system
- If we use the mathematics of probability theory to express all forms of uncertainty and noise associated with our model...
- ...then *inverse probability* (i.e. Bayes rule) allows us to infer unknown quantities, adapt our models, make predictions and learn from data.

Bayesian Machine Learning

Everything follows from two simple rules:

Sum rule: $P(x) = \sum_y P(x, y)$

Product rule: $P(x, y) = P(x)P(y|x)$

$$P(\theta|\mathcal{D}, m) = \frac{P(\mathcal{D}|\theta, m)P(\theta|m)}{P(\mathcal{D}|m)}$$

$P(\mathcal{D}\|\theta, m)$	likelihood of parameters θ in model m
$P(\theta\|m)$	prior probability of θ
$P(\theta\|\mathcal{D}, m)$	posterior of θ given data $\mathcal{D}$

Prediction:

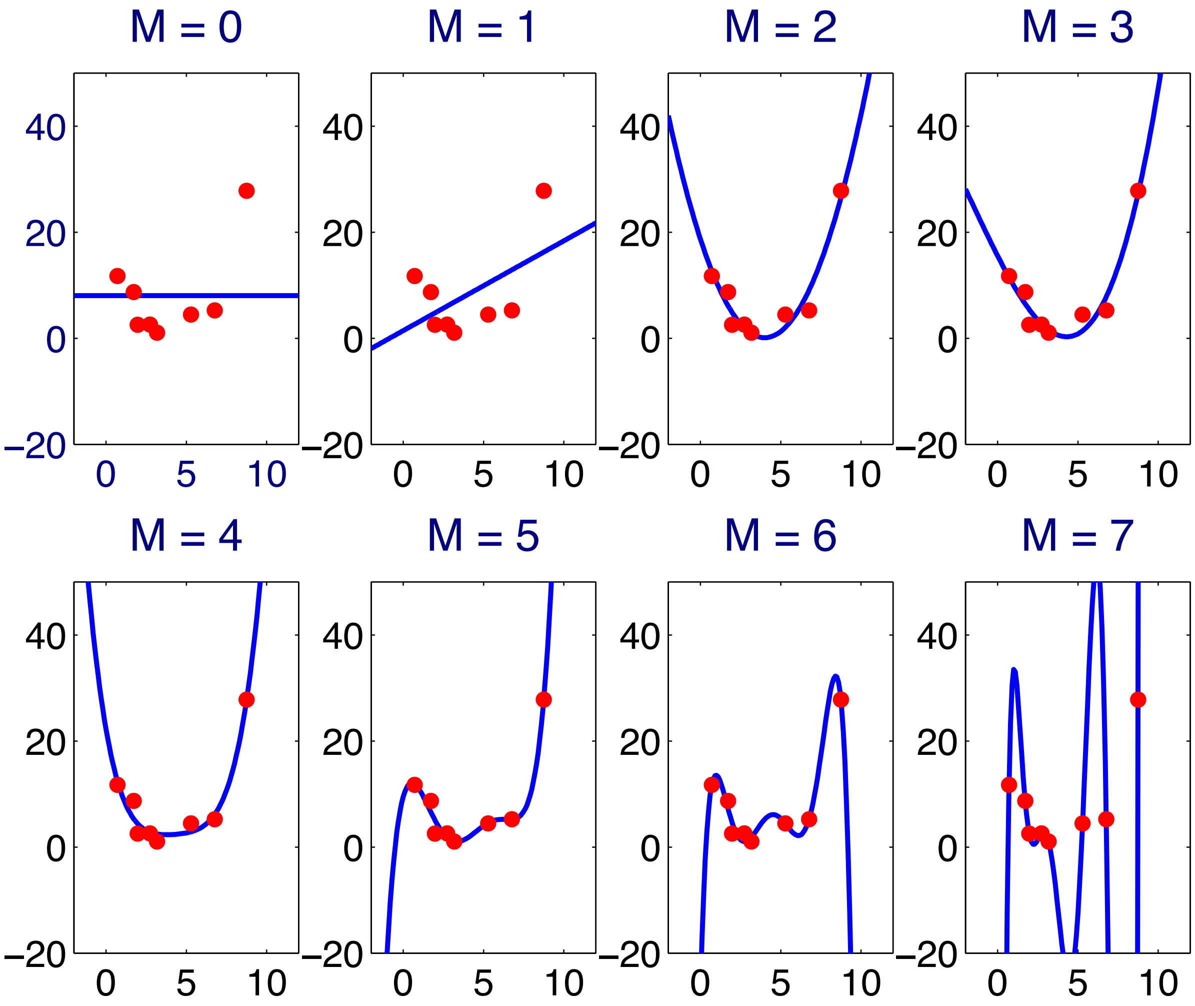
$$P(x|\mathcal{D}, m) = \int P(x|\theta, \mathcal{D}, m)P(\theta|\mathcal{D}, m)d\theta$$

Model Comparison:

$$P(m|\mathcal{D}) = \frac{P(\mathcal{D}|m)P(m)}{P(\mathcal{D})}$$

$$P(\mathcal{D}|m) = \int P(\mathcal{D}|\theta, m)P(\theta|m)\,d\theta$$

Model Comparison



Bayesian Occam's Razor and Model Comparison

Compare model classes, e.g. m and m', using posterior probabilities given $\mathcal{D}$:

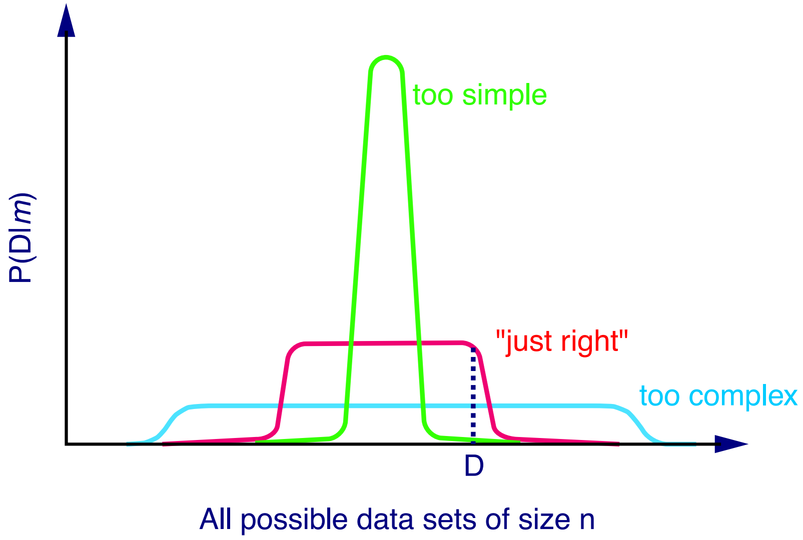
$$p(m|\mathcal{D}) = \frac{p(\mathcal{D}|m)\,p(m)}{p(\mathcal{D})}, \qquad p(\mathcal{D}|m) = \int p(\mathcal{D}|\boldsymbol{\theta}, m)\; p(\boldsymbol{\theta}|m)\; d\boldsymbol{\theta}$$

Interpretations of the Marginal Likelihood ("model evidence"):

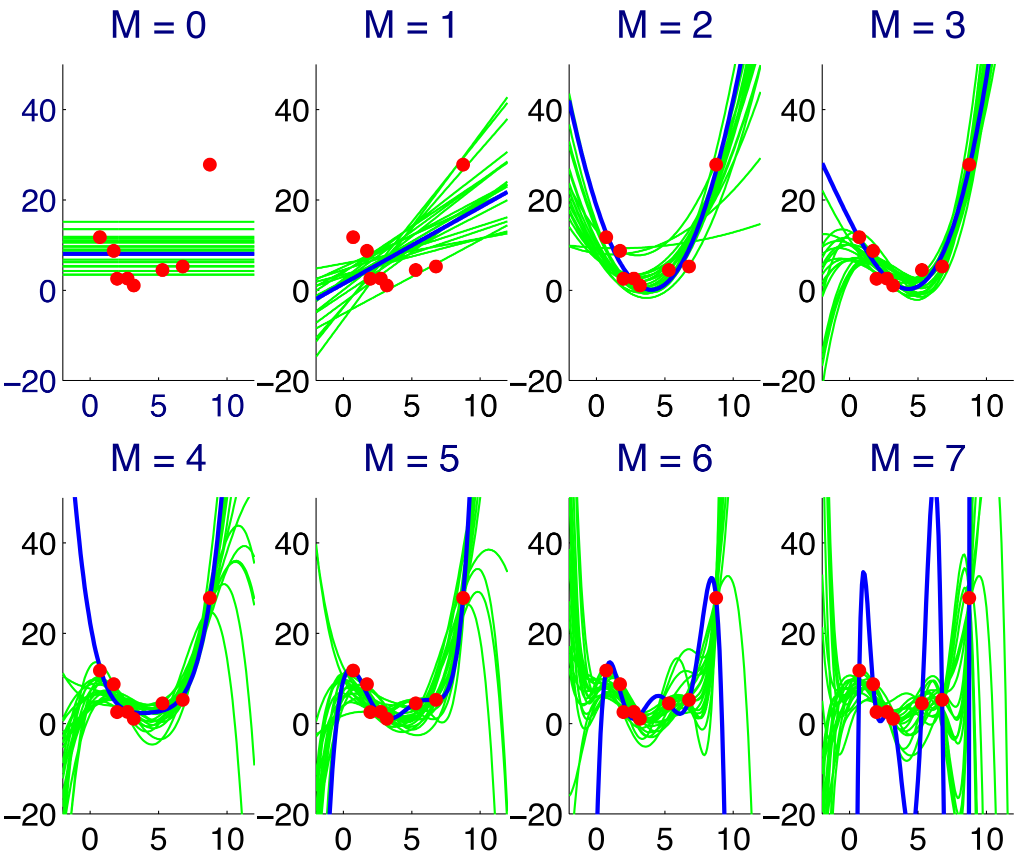
- The probability that *randomly selected* parameters from the prior would generate $\mathcal{D}$.
- Probability of the data under the model, *averaging* over all possible parameter values.
- $\log_2\left(\frac{1}{p(\mathcal{D}|m)}\right)$ is the number of *bits of surprise* at observing data $\mathcal{D}$ under model m.

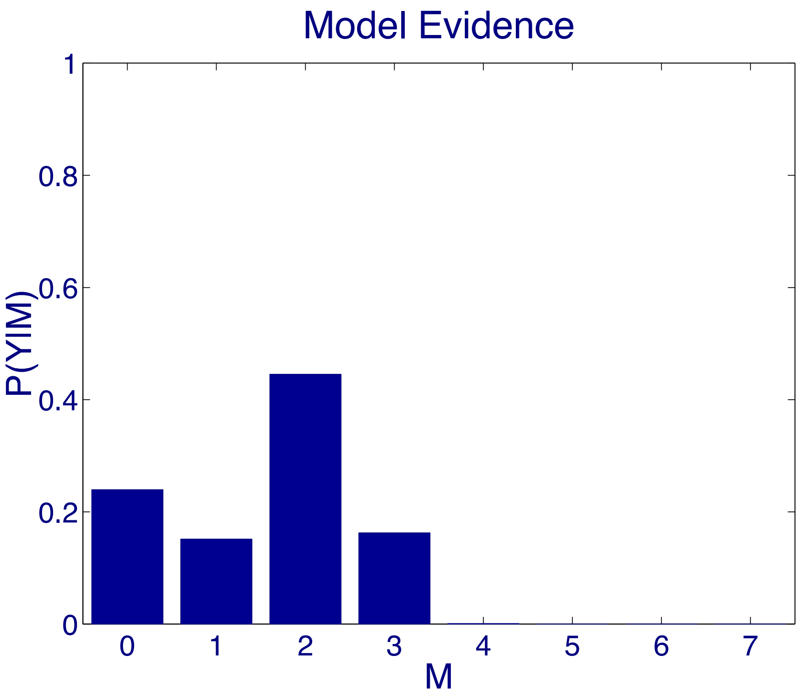
Model classes that are too simple are unlikely to generate the data set.

Model classes that are too complex can generate many possible data sets, so again, they are unlikely to generate that particular data set at random.



Bayesian Model Comparison: Occam's Razor at Work





For example, for quadratic polynomials ($m = 2$): $y = a_0 + a_1 x + a_2 x^2 + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma^2)$ and parameters $\boldsymbol{\theta} = (a_0\ a_1\ a_2\ \sigma)$

```
demo:  polybayes
```

Parametric vs Nonparametric Models

- *Parametric models* assume some finite set of parameters θ. Given the parameters, future predictions, x, are independent of the observed data, $\mathcal{D}$:

$$P(x|\theta, \mathcal{D}) = P(x|\theta)$$

  therefore θ capture everything there is to know about the data.

- So the complexity of the model is bounded even if the amount of data is unbounded. This makes them not very flexible.

- *Non-parametric models* assume that the data distribution cannot be defined in terms of such a finite set of parameters. But they can often be defined by assuming an *infinite dimensional* θ. Usually we think of θ as a *function*.

- The amount of information that θ can capture about the data $\mathcal{D}$ can grow as the amount of data grows. This makes them more flexible.

Bayesian nonparametrics

A simple framework for modelling complex data.

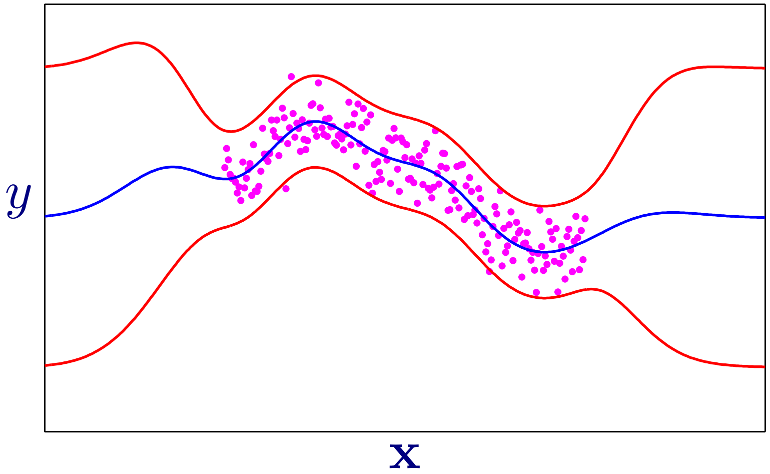
Nonparametric models can be viewed as having infinitely many parameters

Examples of non-parametric models:

Parametric	Non-parametric	Application
polynomial regression	Gaussian processes	function approx.
logistic regression	Gaussian process classifiers	classification
mixture models, k-means	Dirichlet process mixtures	clustering
hidden Markov models	infinite HMMs	time series
factor analysis / pPCA / PMF	infinite latent factor models	feature discovery
...		

Nonlinear regression and Gaussian processes

Consider the problem of nonlinear regression:
You want to learn a function f with error bars from data $\mathcal{D} = \{\mathbf{X}, \mathbf{y}\}$



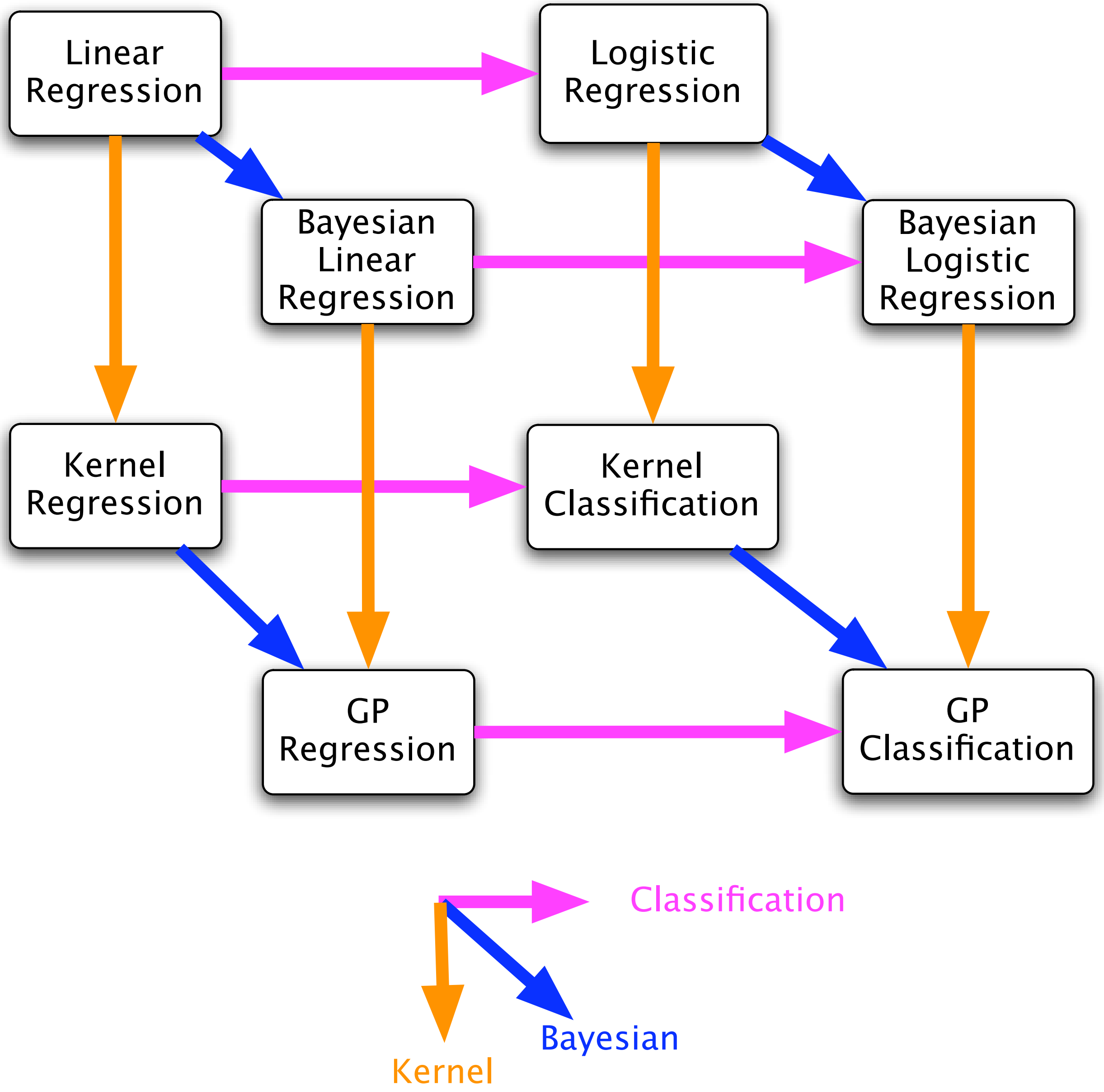
A Gaussian process defines a distribution over functions $p(f)$ which can be used for Bayesian regression:

$$p(f|\mathcal{D}) = \frac{p(f)p(\mathcal{D}|f)}{p(\mathcal{D})}$$

Let $\mathbf{f} = (f(x_1), f(x_2), \ldots, f(x_n))$ be an n-dimensional vector of function values evaluated at n points $x_i \in \mathcal{X}$. Note, $\mathbf{f}$ is a random variable.

Definition: $p(f)$ is a Gaussian process if for *any* finite subset $\{x_1, \ldots, x_n\} \subset \mathcal{X}$, the marginal distribution over that subset $p(\mathbf{f})$ is multivariate Gaussian.

A picture


Linear Regression
Logistic Regression
Bayesian Linear Regression
Bayesian Logistic Regression
Kernel Regression
Kernel Classification
GP Regression
GP Classification
Classification
Bayesian
Kernel

Gaussian process covariance functions (kernels)

$p(f)$ is a Gaussian process if for *any* finite subset $\{x_1, \ldots, x_n\} \subset \mathcal{X}$, the marginal distribution over that finite subset $p(\mathbf{f})$ has a multivariate Gaussian distribution.

Gaussian processes (GPs) are parameterized by a mean function, $\mu(x)$, and a covariance function, or kernel, $K(x, x')$.

$$p(f(x), f(x')) = \mathrm{N}(\boldsymbol{\mu}, \Sigma)$$

where

$$\boldsymbol{\mu} = \begin{bmatrix} \mu(x) \\ \mu(x') \end{bmatrix} \quad \Sigma = \begin{bmatrix} K(x,x) & K(x,x') \\ K(x',x) & K(x',x') \end{bmatrix}$$

and similarly for $p(f(x_1), \ldots, f(x_n))$ where now $\boldsymbol{\mu}$ is an $n \times 1$ vector and Σ is an $n \times n$ matrix.

Gaussian process covariance functions

Gaussian processes (GPs) are parameterized by a mean function, $\mu(x)$, and a covariance function, $K(x, x')$, where $\mu(x) = \mathsf{E}(f(x))$ and $K(x, x') = \mathsf{Cov}(f(x), f(x'))$.

An example covariance function:

$$K(x, x') = v_0 \exp\left\{ -\left(\frac{|x - x'|}{r} \right)^{\alpha} \right\} + v_1 + v_2\, \delta_{ij}$$

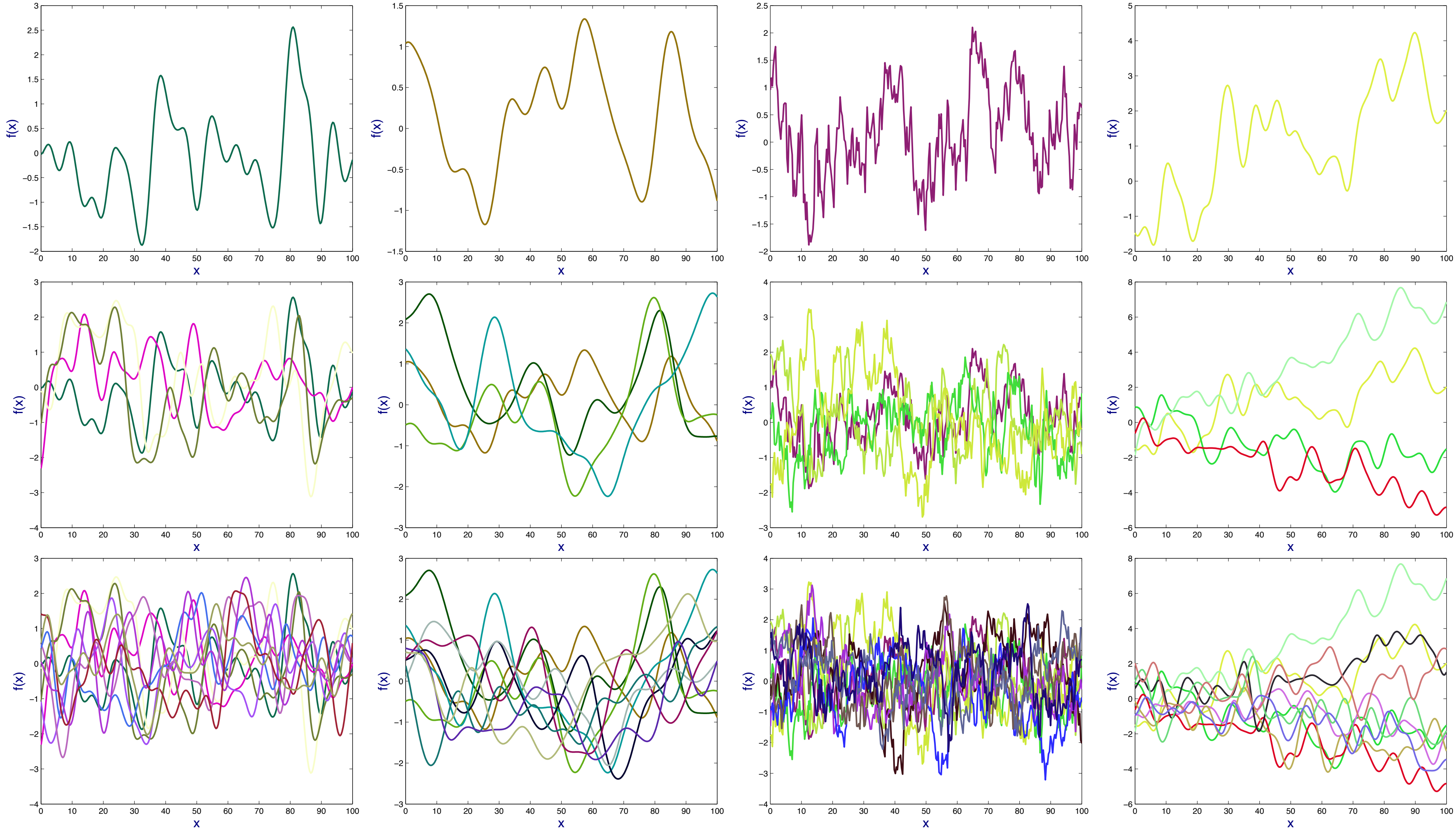
with parameters $(v_0, v_1, v_2, r, \alpha)$.

These kernel parameters are interpretable and can be learned from data:

v_0	signal variance
v_1	variance of bias
v_2	noise variance
r	lengthscale
α	roughness

Once the mean and covariance functions are defined, everything else about GPs follows from the basic rules of probability applied to mutivariate Gaussians.

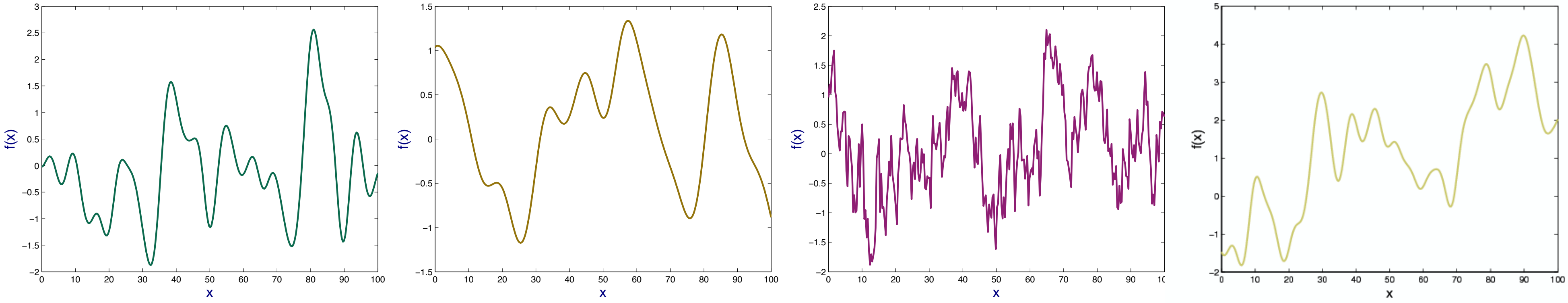
Samples from GPs with different $K(x, x')$



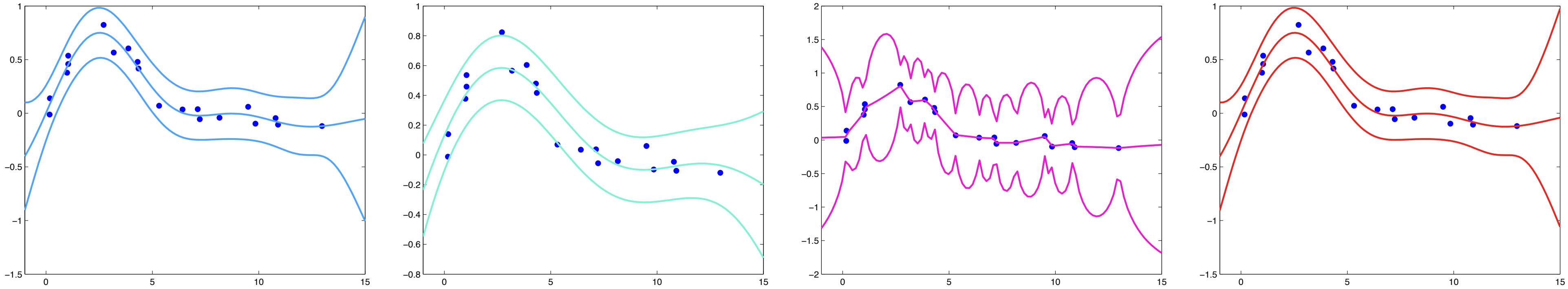
gpdemogen

Prediction using GPs with different $K(x, x')$

A sample from the prior for each covariance function:



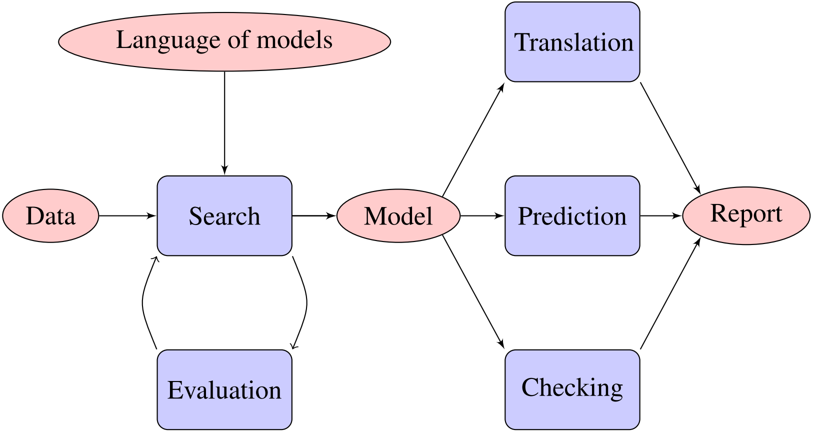
Corresponding predictions, mean with two standard deviations:



`gpdemo`

The Automatic Statistician

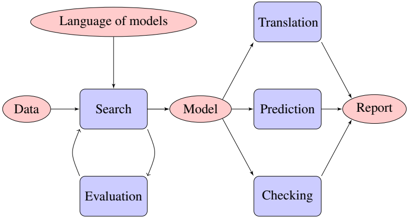
What would an automatic statistician do?



GOALS OF THE AUTOMATIC STATISTICIAN PROJECT

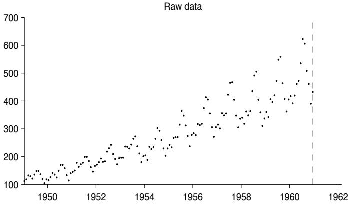
- Provide a set of tools for understanding data that require minimal expert input
- Uncover challenging research problems in e.g.
    - Automated inference
    - Model construction and comparison
    - Data visualisation and interpretation
- Advance the field of machine learning in general

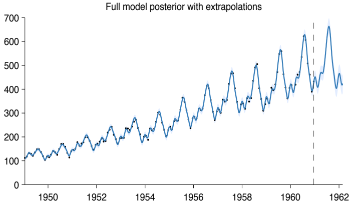
INGREDIENTS OF AN AUTOMATIC STATISTICIAN



- **An open-ended language of models**
  - Expressive enough to capture real-world phenomena…
  - …and the techniques used by human statisticians
- **A search procedure**
  - To efficiently explore the language of models
- **A principled method of evaluating models**
  - Trading off complexity and fit to data
- **A procedure to automatically explain the models**
  - Making the assumptions of the models explicit…
  - …in a way that is intelligible to non-experts

PREVIEW: AN ENTIRELY AUTOMATIC ANALYSIS

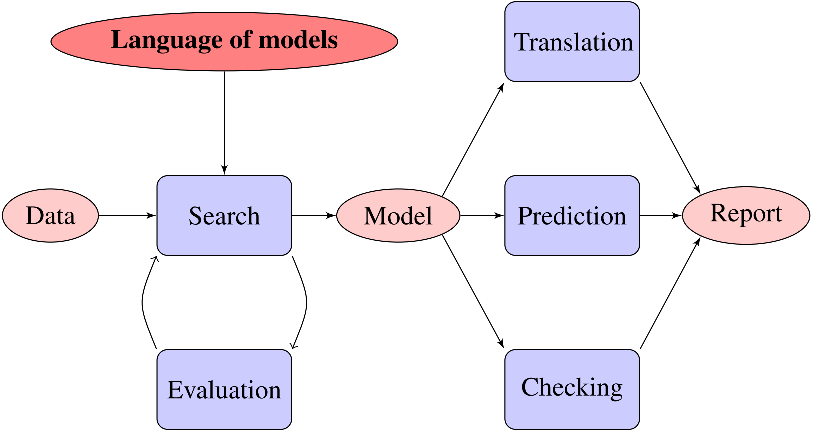




Four additive components have been identified in the data

- 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.

DEFINING A LANGUAGE OF MODELS



DEFINING A LANGUAGE OF REGRESSION MODELS

- Regression consists of **learning a function** $f : \mathcal{X} \to \mathcal{Y}$ from inputs to outputs from example input / output pairs

- Language should include **simple parametric forms**...
  - e.g. Linear functions, Polynomials, Exponential functions

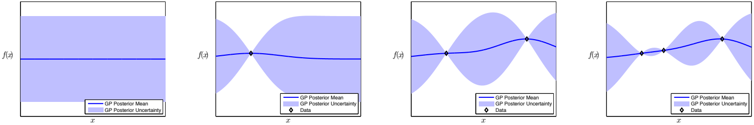
- ...as well as functions specified by **high level properties**
  - e.g. Smoothness, Periodicity

- Inference should be **tractable for all models** in language

We can build regression models with Gaussian processes

- GPs are distributions over functions such that any finite subset of function evaluations, $(f(x_1), f(x_2), \ldots f(x_N))$, have a joint Gaussian distribution

- A GP is completely specified by
  - Mean function, $\mu(x) = \mathbb{E}(f(x))$
  - Covariance / kernel function, $k(x, x') = \text{Cov}(f(x), f(x'))$
  - Denoted $f \sim \text{GP}(\mu, k)$



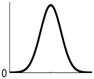
A LANGUAGE OF GAUSSIAN PROCESS KERNELS

- It is common practice to use a zero mean function since the mean can be marginalised out
  - Suppose, $f(x) \mid a \sim \text{GP}(a \times \mu(x), k(x, x'))$ where $a \sim \mathcal{N}(0, 1)$
  - Then equivalently, $f(x) \sim \text{GP}(0, \mu(x)\mu(x') + k(x, x'))$

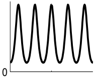
- We therefore define a language of GP regression models by specifying a **language of kernels**

THE ATOMS OF OUR LANGUAGE

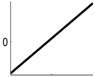
Five base kernels



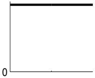
Squared exp. (SE)



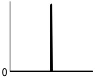
Periodic (PER)



Linear (LIN)

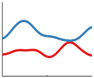


Constant (C)

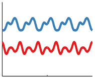


White noise (WN)

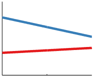
Encoding for the following types of functions



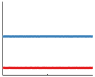
Smooth functions



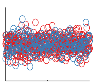
Periodic functions



Linear functions



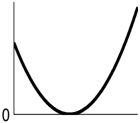
Constant functions



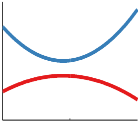
Gaussian noise

The composition rules of our language

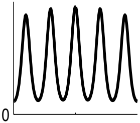
- Two main operations: addition, multiplication



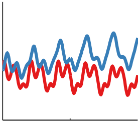
Lin × Lin



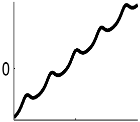
quadratic functions



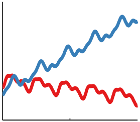
SE × Per



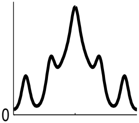
locally periodic



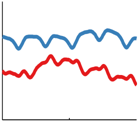
Lin + Per



periodic plus linear trend



SE + Per



periodic plus smooth trend

Modeling changepoints

Assume $f_1(x) \sim GP(0, k_1)$ and $f_2(x) \sim GP(0, k_2)$. Define:

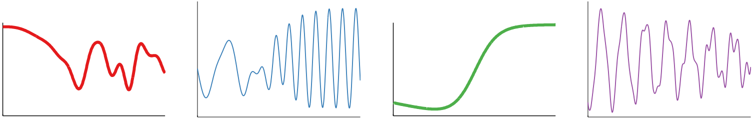
$$f(x) = (1 - \sigma(x))\, f_1(x) + \sigma(x)\, f_2(x)$$

where σ is a sigmoid function between 0 and 1.

Then $f \sim GP(0, k)$, where

$$k(x, x') = (1 - \sigma(x))\, k_1(x, x')\, (1 - \sigma(x')) + \sigma(x)\, k_2(x, x')\, \sigma(x')$$

We define the changepoint operator $k = \mathrm{CP}(k_1, k_2)$.

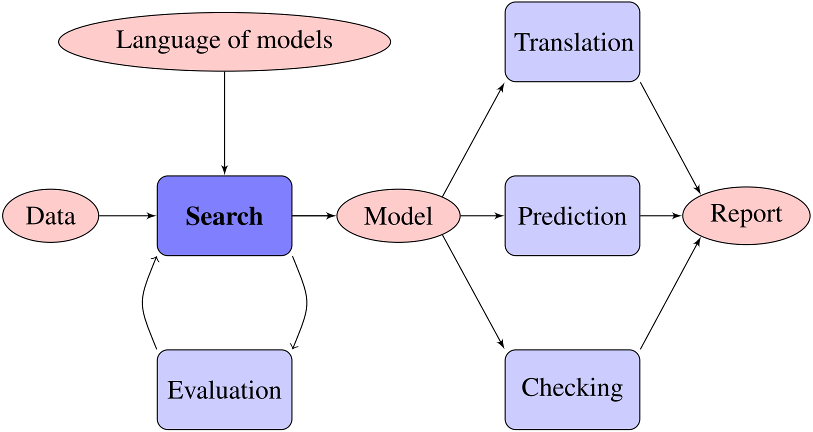


AN EXPRESSIVE LANGUAGE OF MODELS

Regression model	Kernel
GP smoothing	$\text{SE} + \text{WN}$
Linear regression	$\text{C} + \text{LIN} + \text{WN}$
Multiple kernel learning	$\sum \text{SE} + \text{WN}$
Trend, cyclical, irregular	$\sum \text{SE} + \sum \text{PER} + \text{WN}$
Fourier decomposition	$\text{C} + \sum \cos + \text{WN}$
Sparse spectrum GPs	$\sum \cos + \text{WN}$
Spectral mixture	$\sum \text{SE} \times \cos + \text{WN}$
Changepoints	e.g. $\text{CP}(\text{SE}, \text{SE}) + \text{WN}$
Heteroscedasticity	e.g. $\text{SE} + \text{LIN} \times \text{WN}$

Note: cos is a special case of our version of PER

Discovering a good model via search



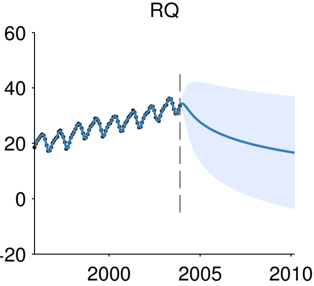
DISCOVERING A GOOD MODEL VIA SEARCH

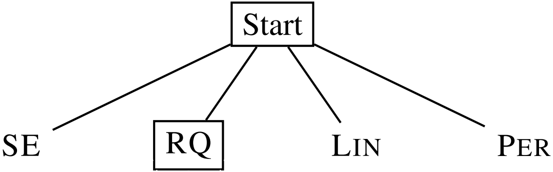
- Language defined as the arbitrary composition of five base kernels (WN, C, LIN, SE, PER) via three operators $(+, \times, \text{CP})$.

- The space spanned by this language is open-ended and can have a high branching factor requiring a judicious search

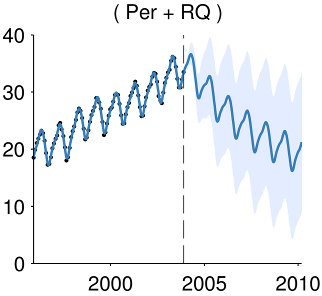
- We propose a greedy search for its simplicity and similarity to human model-building

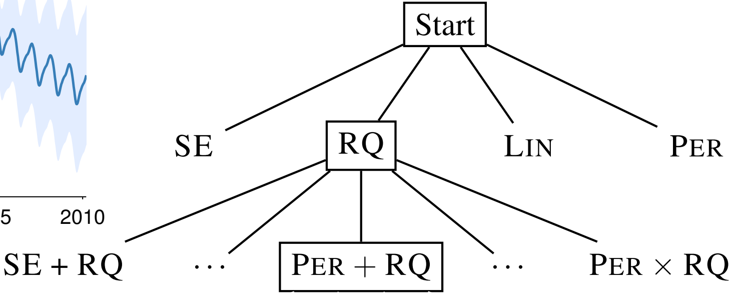
EXAMPLE: MAUNA LOA KEELING CURVE





EXAMPLE: MAUNA LOA KEELING CURVE





Start

SE

RQ

LIN

PER

SE + RQ

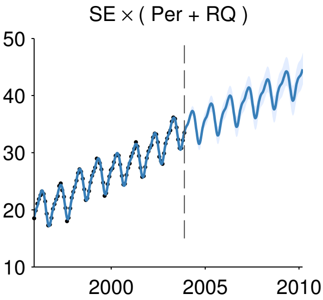
…

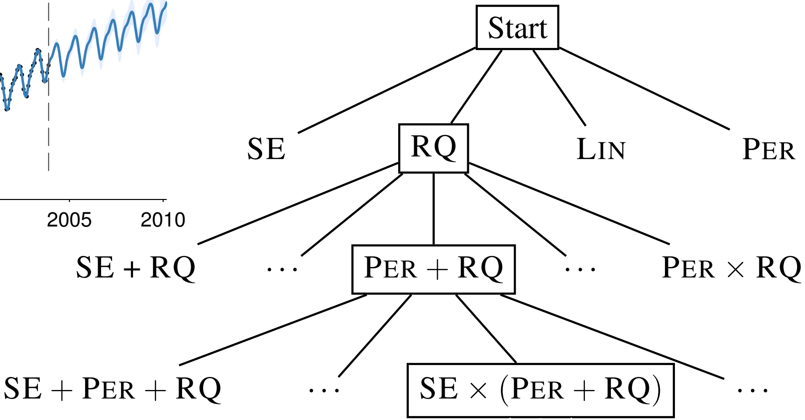
PER + RQ

…

PER × RQ

Example: Mauna Loa Keeling Curve





Start

SE

RQ

Lin

Per

SE + RQ

· · ·

Per + RQ

· · ·

Per × RQ

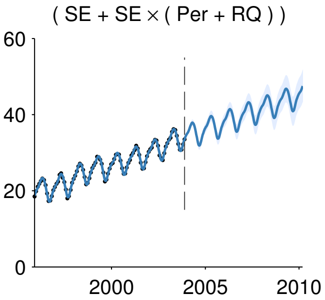
SE + Per + RQ

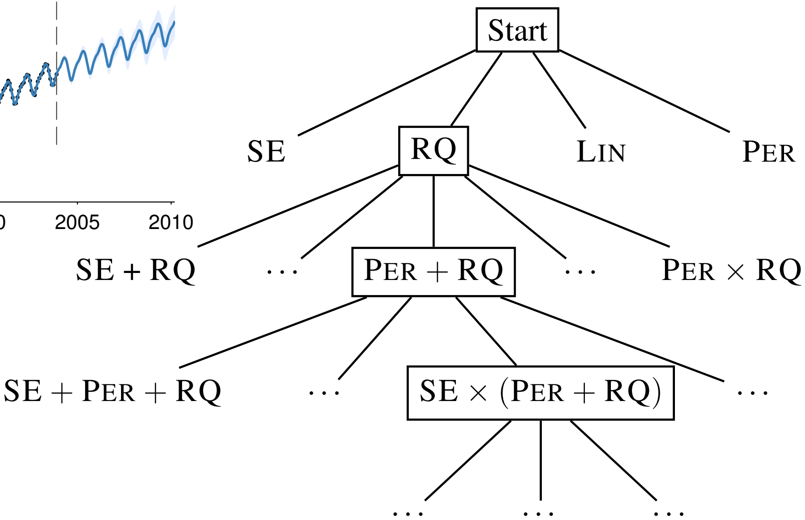
· · ·

SE × (Per + RQ)

· · ·

Example: Mauna Loa Keeling Curve





Start

SE

RQ

Lin

Per

SE + RQ

...

Per + RQ

...

Per × RQ

SE + Per + RQ

...

SE × (Per + RQ)

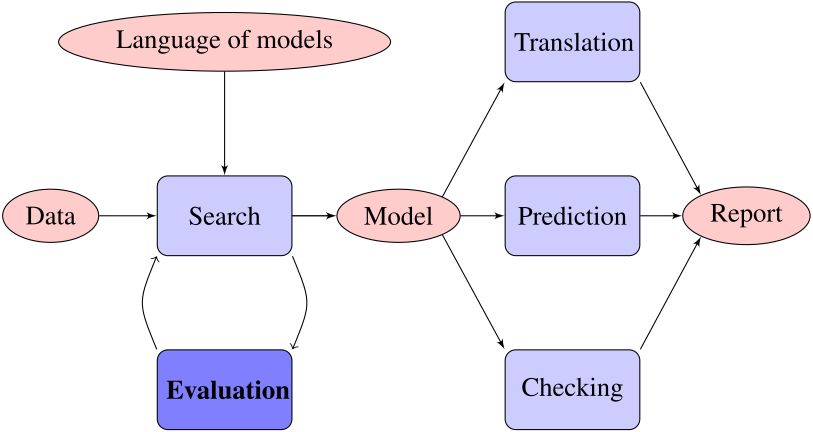
...

...

...

...

MODEL EVALUATION



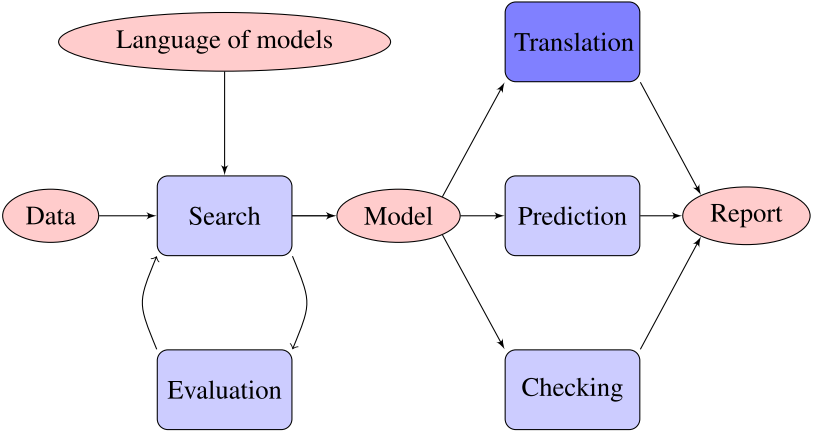
Model evaluation

- After proposing a new model its kernel parameters are optimised by conjugate gradients
- We evaluate each optimised model, M, using the model evidence (marginal likelihood) which can be computed analytically for GPs
- We penalise the marginal likelihood for the optimised kernel parameters using the Bayesian Information Criterion (BIC):

$$-0.5 \times \mathrm{BIC}(M) = \log p(D \mid M) - \frac{p}{2} \log n$$

where p is the number of kernel parameters, D represents the data, and n is the number of data points.

AUTOMATIC TRANSLATION OF MODELS



AUTOMATIC TRANSLATION OF MODELS

- Search can produce **arbitrarily complicated models** from open-ended language but two main properties allow description to be automated

- Kernels can be **decomposed** into a **sum of products**
  - A sum of kernels corresponds to a sum of functions
  - Therefore, we can describe each product of kernels separately

- Each kernel in a product modifies a model in a **consistent** way
  - Each kernel roughly corresponds to an *adjective*

SUM OF PRODUCTS NORMAL FORM

Suppose the search finds the following kernel

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

SUM OF PRODUCTS NORMAL FORM

Suppose the search finds the following kernel

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

The changepoint can be converted into a sum of products

$$\text{SE} \times (\text{WN} \times \textsc{Lin} + \text{C} \times \boldsymbol{\sigma} + \textsc{Per} \times \bar{\boldsymbol{\sigma}})$$

SUM OF PRODUCTS NORMAL FORM

Suppose the search finds the following kernel

$$\text{SE} \times (\text{WN} \times \text{LIN} + \text{CP}(\text{C}, \text{PER}))$$

The changepoint can be converted into a sum of products

$$\text{SE} \times (\text{WN} \times \text{LIN} + \text{C} \times \boldsymbol{\sigma} + \text{PER} \times \bar{\boldsymbol{\sigma}})$$

Multiplication can be distributed over addition

$$\text{SE} \times \text{WN} \times \text{LIN} + \text{SE} \times \text{C} \times \boldsymbol{\sigma} + \text{SE} \times \text{PER} \times \bar{\boldsymbol{\sigma}}$$

Sum of products normal form

Suppose the search finds the following kernel

$$\mathrm{SE} \times (\mathrm{WN} \times \textsc{Lin} + \mathrm{CP}(\mathrm{C}, \textsc{Per}))$$

The changepoint can be converted into a sum of products

$$\mathrm{SE} \times (\mathrm{WN} \times \textsc{Lin} + \mathrm{C} \times \boldsymbol{\sigma} + \textsc{Per} \times \bar{\boldsymbol{\sigma}})$$

Multiplication can be distributed over addition

$$\mathrm{SE} \times \mathrm{WN} \times \textsc{Lin} + \mathrm{SE} \times \mathrm{C} \times \boldsymbol{\sigma} + \mathrm{SE} \times \textsc{Per} \times \bar{\boldsymbol{\sigma}}$$

Simplification rules are applied

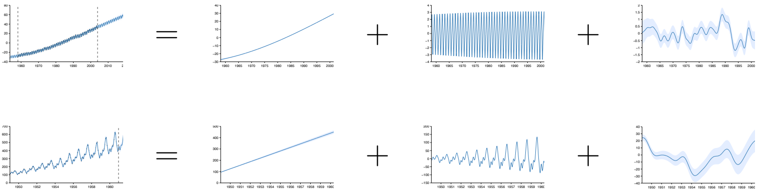
$$\mathrm{WN} \times \textsc{Lin} + \mathrm{SE} \times \boldsymbol{\sigma} + \mathrm{SE} \times \textsc{Per} \times \bar{\boldsymbol{\sigma}}$$

SUMS OF KERNELS ARE SUMS OF FUNCTIONS

If $f_1 \sim \mathrm{GP}(0, k_1)$ and independently $f_2 \sim \mathrm{GP}(0, k_2)$ then

$$f_1 + f_2 \sim \mathrm{GP}(0, k_1 + k_2)$$

e.g.

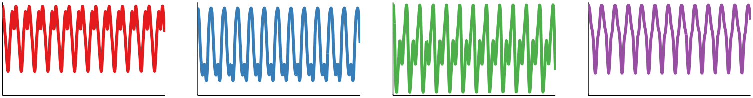


We can therefore describe each component separately

PRODUCTS OF KERNELS

$$\underbrace{\text{PER}}_{\text{periodic function}}$$

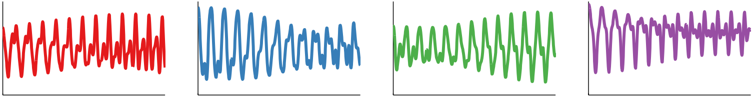
On their own, each kernel is described by a standard noun phrase



Products of kernels - SE

$$\underbrace{\text{SE}}_{\text{approximately}} \times \underbrace{\text{Per}}_{\text{periodic function}}$$

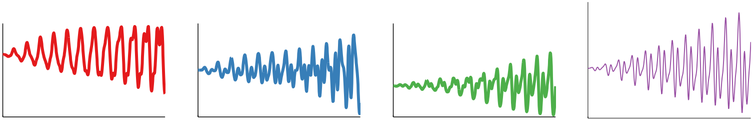
Multiplication by SE removes long range correlations from a model since $\text{SE}(x, x')$ decreases monotonically to 0 as $|x - x'|$ increases.



PRODUCTS OF KERNELS - LIN

$$\underbrace{\text{SE}}_{\text{approximately}} \times \underbrace{\text{PER}}_{\text{periodic function}} \times \underbrace{\text{LIN}}_{\text{with linearly growing amplitude}}$$

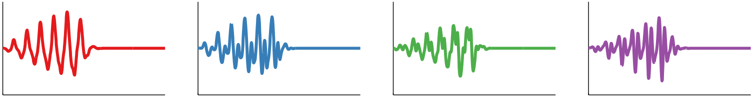
Multiplication by LIN is equivalent to multiplying the function being modeled by a linear function. If $f(x) \sim \text{GP}(0, k)$, then $xf(x) \sim \text{GP}(0, k \times \text{LIN})$. This causes the standard deviation of the model to vary linearly without affecting the correlation.



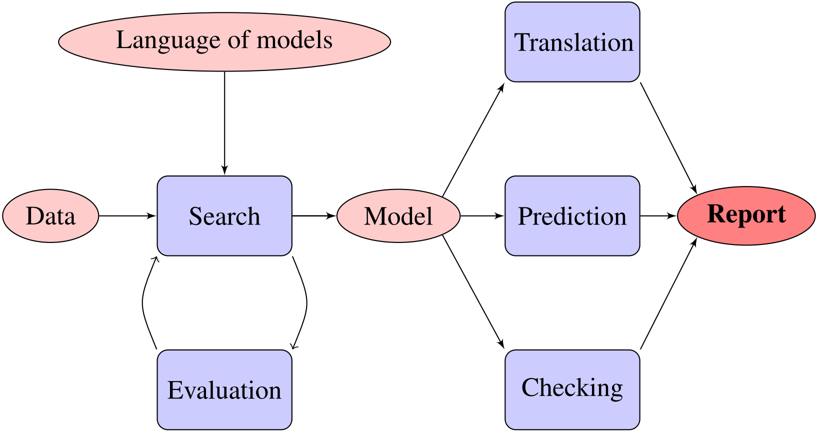
Products of kernels - changepoints

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

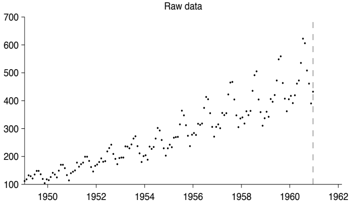
Multiplication by $\boldsymbol{\sigma}$ is equivalent to multiplying the function being modeled by a sigmoid.

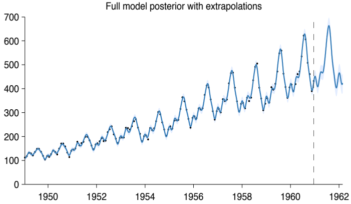


AUTOMATICALLY GENERATED REPORTS



EXAMPLE: AIRLINE PASSENGER VOLUME


Raw data

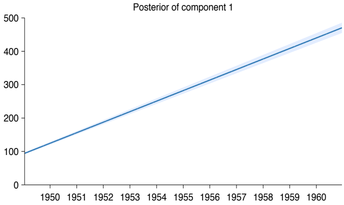

Full model posterior with extrapolations

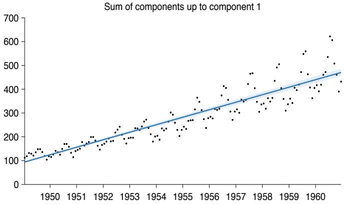
Four additive components have been identified in the data

- 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.

Example: Airline passenger volume

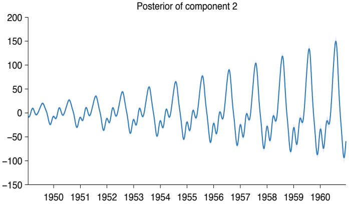
This component is linearly increasing.

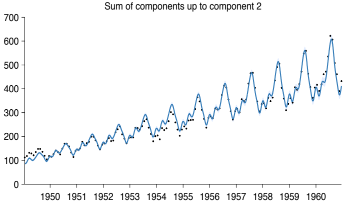




EXAMPLE: AIRLINE PASSENGER VOLUME

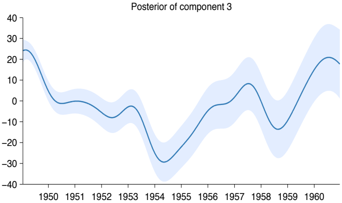
This component is approximately periodic with a period of 1.0 years and varying amplitude. Across periods the shape of this function varies very smoothly. The amplitude of the function increases linearly. The shape of this function within each period has a typical lengthscale of 6.0 weeks.

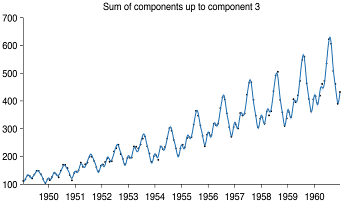




Example: Airline passenger volume

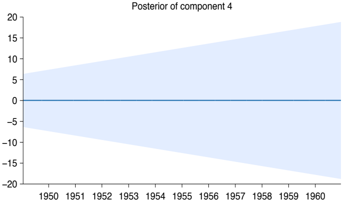
This component is a smooth function with a typical lengthscale of 8.1 months.

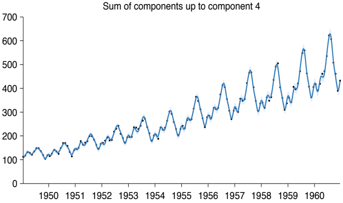




EXAMPLE: AIRLINE PASSENGER VOLUME

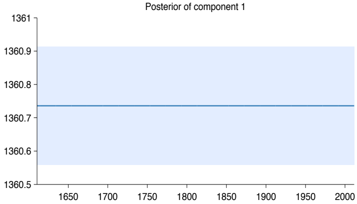
This component models uncorrelated noise. The standard deviation of the noise increases linearly.

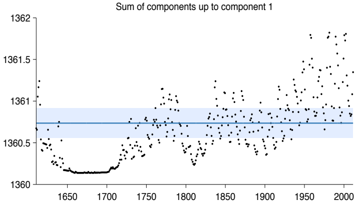

Posterior of component 4


Sum of components up to component 4

EXAMPLE: SOLAR IRRADIANCE

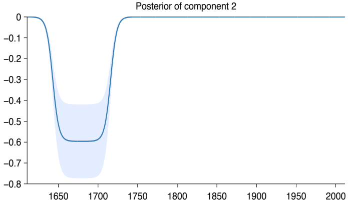
This component is constant.

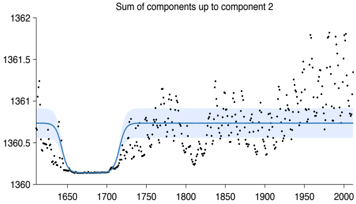




EXAMPLE: SOLAR IRRADIANCE

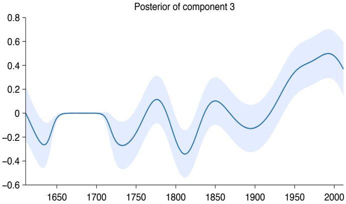
This component is constant. This component applies from 1643 until 1716.

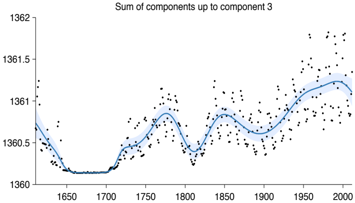




EXAMPLE: SOLAR IRRADIANCE

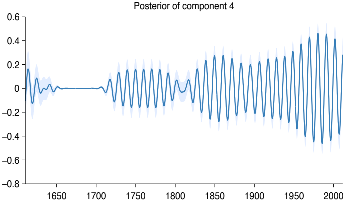
This component is a smooth function with a typical lengthscale of 23.1 years. This component applies until 1643 and from 1716 onwards.

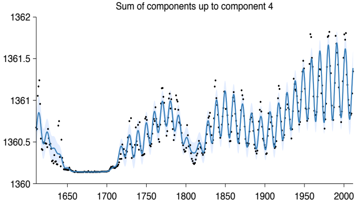

Posterior of component 3


Sum of components up to component 3

EXAMPLE: SOLAR IRRADIANCE

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.



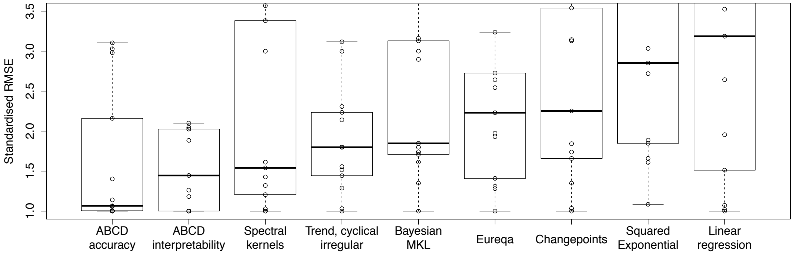


OTHER EXAMPLES

See `http://www.automaticstatistician.com`

Good predictive performance as well

Standardised RMSE over 13 data sets



- Tweaks can be made to the algorithm to improve accuracy or interpretability of models produced…

- …but both methods are *highly competitive* at extrapolation (shown above) and interpolation

Model Checking and Criticism

- Good statistical modelling should include model criticism:
  - Does the data match the assumptions of the model?
  - For example, if the model assumed Gaussian noise, does a Q-Q plot reveal non-Gaussian residuals?
- Our automatic statistician does posterior predictive checks, dependence tests and residual tests
- We have also been developing more systematic nonparametric approaches to model criticism using kernel two-sample testing with MMD.

Lloyd, J. R., and Ghahramani, Z. (2014) Statistical Model Criticism using Kernel Two Sample Tests. http://mlg.eng.cam.ac.uk/Lloyd/papers/kernel-model-checking.pdf

CHALLENGES

- Interpretability / accuracy
- Increasing the expressivity of language
  - e.g. Monotonocity, positive functions, symmetries
- Computational complexity of searching through a huge space of models
- Extending the automatic reports to multidimensional datasets
  - Search and descriptions naturally extend to multiple dimensions, but automatically generating relevant visual summaries harder

CURRENT AND FUTURE DIRECTIONS

- Automatic statistician for:
  - * One-dimensional time series
  - * Linear regression (classical)
  - Multivariate nonlinear regression (c.f. Duvenaud, Lloyd et al, ICML 2013)
  - Multivariate classification (w/ Nikola Mrksic)
  - Completing and interpreting tables and databases (w/ Kee Chong Tan)

- Probabilistic programming
  - Probabilistic models are expressed in a general (Turing complete) programming language
  - A universal inference engine can then be used to infer unobserved variables given observed data
  - This can be used to implement seach over the model space in an automatic statistician

Summary

- We have presented the beginnings of an automatic statistician
- Our system
  - Defines an open-ended language of models
  - Searches greedily through this space
  - Produces detailed reports describing patterns in data
  - Performs automatic model criticism
- Extrapolation and interpolation performance highly competitive
- We believe this line of research has the potential to make powerful statistical model-building techniques accessible to non-experts

REFERENCES

Website: http://www.automaticstatistician.com

Duvenaud, D., Lloyd, J. R., Grosse, R., Tenenbaum, J. B. and Ghahramani, Z. (2013) Structure Discovery in Nonparametric Regression through Compositional Kernel Search. ICML 2013.

Lloyd, J. R., Duvenaud, D., Grosse, R., Tenenbaum, J. B. and Ghahramani, Z. (2014) Automatic Construction and Natural-language Description of Nonparametric Regression Models AAAI 2014. http://arxiv.org/pdf/1402.4304v2.pdf

Lloyd, J. R., and Ghahramani, Z. (2014) Statistical Model Criticism using Kernel Two Sample Tests http://mlg.eng.cam.ac.uk/Lloyd/papers/kernel-model-checking.pdf

Ghahramani, Z. (2013) Bayesian nonparametrics and the probabilistic approach to modelling *Philosophical Trans. Royal Society A* 371: 20110553.

Looking for postdocs!