STA414/2104: Week 11

Gaussian Process Models

Alex Stringer

March 27th, 2018

Citation

Material in these slides was adapted from

- Previous course slides by Radford Neal
- Previous course slides by Ruslan Salakhutdinov
- David Duvenaud's PhD thesis: https://raw.githubusercontent.com/duvenaud/phd-thesis/master/kernels.pdf
- Rasmussen and Williams (2006): *Gaussian Processes for Machine Learning*

Recap

So far in this course we have talked about

- Building probabilistic models for data, mathematically
- Fitting probabilistic models to data
- Comparing models

We have talked about theoretical and practical considerations with regard to these topics

Linear Basis Function Models

Recall the linear basis function model,

$$y_i = \sum_{m=1}^{M} \beta_m \phi(\mathbf{x}_i) + \epsilon_i, \epsilon_i \sim N(0, \sigma^2)$$

where the $(y_i, \mathbf{x}_i)$ represents our training set.

The $\phi(\cdot)$ are basis functions, which can be pretty much anything- identity, constant, Gaussian, step functions... how to choose?

How to choose M?

This week

This week, we will investigate this question theoretically, leading to **Gaussian Process Models**.

We will then talk about how this relates to choosing basis functions in practice (sort of- we'll actually choose Kernels).

Don't get your hopes up though. We're not going to solve the problem fully, yielding a magic formula for choosing kernels/basis functions. We'll characterise the problem and discuss how to approach it, but at the end, it's still the case that the more time and effort you put in to understanding your data (and the fit of different models to it), the better your final model will be.

Bayesian LBFM

First, let's talk about how to choose M. If we had a particular family of basis functions in mind, we could choose by cross-validation.

What happens if we make M really big? Like, bigger than n?

Regularization of some kind is necessary to estimate such a model, as it has more parameters than training observations.

Since, in specifying basis functions to begin with, we are really trying to encode our *prior beliefs* about the structure in the data, and we need a form of regularization, let's use a Bayesian approach.

For fixed M, put a prior on $\beta = (\beta_1, \ldots, \beta_M)$:

$$\beta \sim N(0, \mathbf{S}_0)$$

Bayesian LBFM

Computing this prior density involves the inversion of a $M \times M$ covariance matrix.

But we said $M > n$. In this case, it is more computationally efficient to work with the training data itself.

Does the prior distribution we placed on β in turn imply some prior distribution on $\mathbf{y}$?

$$\begin{aligned} \beta &\sim N(0, \mathbf{S}_0) \\ \mathbf{y} &= \mathbf{\Phi}(\mathbf{X})\beta + \boldsymbol{\epsilon} \\ \boldsymbol{\epsilon} &\sim N(0, \sigma^2 \mathbf{I}) \end{aligned}$$

Bayesian LBFM

β and $\boldsymbol{\epsilon}$ are (assumed) independent, and so are jointly normal.

Linear combinations of jointly normal random variables are jointly normal.

$$y \sim N(0, \mathbf{\Phi S \Phi}' + \sigma^2 \mathbf{I})$$

In the common case that $\mathbf{S}_0 = diag(\psi_1^2, \ldots, \psi_M^2)$ (we don't have prior reason to believe that elements of β are correlated):

$$Cov(y_i, y_{i'}) = \sigma^2 I(i = i') + \sum_{m=1}^{M} \psi_m^2 \phi_m(\mathbf{x}_i)\phi_m(\mathbf{x}_{i'})$$

Connection to Latent Variables

We specified correlation in our responses by putting a prior distribution on β.

In lecture 8, we said that a good way to do this was to add latent variables into the model.

These are closely connected: β is just a latent variable, on which we put a probability distribution.

The Bayesian LBFM is a mixed model with the constant part of the intercept taken to be 0.

Parameter Estimation

Parameter estimation can then proceed using the marginal likelihood of $\mathbf{y}$, which can be viewed as

- The "model evidence" from the pure-Bayesian perspective, obtained by integrating out all parameters in the unnormalized posterior (the denominator of the posterior distribution obtained using Bayes' rule)
- The regular old likelihood we would have got from directly specifying the above regression model.

The marginal likelihood depends on σ^2, ψ_m^2, and any additional parameters inside the $\phi_m(\cdot)$ functions. It can be maximized in the usual ways (least squares, gradient descent).

Predictions

So the prior covariance of $\mathbf{y}$ is $\mathbf{\Phi S \Phi'} + \sigma^2 \mathbf{I} \in \mathbb{R}^{n \times n}$.

When $M > n$, it is more computationally attractive to work directly with $\mathbf{y}$.

Can we obtain the predictive distribution of a new (test) point y^*, conditional on the training data?

Predictions

Let

$$\begin{aligned}\mathbf{C} &= Var(\mathbf{y}) \in \mathbb{R}^{n\times n}\\ v &= Var(y^*) \in \mathbb{R}\\ \mathbf{k} &= (Cov(y_1, y^*), \ldots, Cov(y_n, y^*))\end{aligned}$$

Then because y^* and $\mathbf{y}$ are jointly normal, standard results for Gaussian conditionals give

$$y^*|y_1, \ldots y_n \sim N\left(\mathbf{k}'\mathbf{C}^{-1}\mathbf{y}, v - \mathbf{k}'\mathbf{C}^{-1}\mathbf{k}\right)$$

We have implicitly integrated over β.

Choosing Basis Functions

How does this at all relate to the choice of basis functions?

Notice that under this framework, the only impact of the basis functions on either the marginal likelihood of the training data or the predictive distribution of a test case given the training data is through the covariance between $(y_i, y_{i\prime})$, and (y_i, y^*).

We can write the covariance between two training cases as

$$Cov(y_i, y_{i\prime}) = \sigma^2 I(i = i') + \sum_{m=1}^{M} \psi_m^2 \phi_m(\mathbf{x}_i)\phi_m(\mathbf{x}_{i'})$$

Choosing Basis Functions

What if we chose ϕ_m and ψ_m^2 such that

$$\lim_{M \to \infty} \sum_{m=1}^{M} \psi_m^2 \phi_m(\mathbf{x}_i) \phi_m(\mathbf{x}_{i'}) < \infty$$

That is, what if we specified a model with infinite M? Does this make sense to do?

Sure. For example, use a Fourier basis, with

$$\phi_{2m-1}(x) = \sin(a_m x); \phi_{2m}(x) = \cos(a_m x); a_m \sim N(0, \rho^2)$$

Or if we scale the features so $x \in (0, 1)$, can use polynomial basis functions.

Interpreting This

To interpret this, write

$$Cov(y_i, y_{i\prime}) = \sigma^2 I(i = i') + K(\mathbf{x}_i, \mathbf{x}_{i'})$$

where

$$K(\mathbf{x}, \mathbf{x}') = \sum_{m=1}^{M} \psi_m^2 \phi_m(\mathbf{x}) \phi_m(\mathbf{x}')$$

is the "noise-free" covariance function, which specifies how correlated y and y' are, given their associated features x, x'.

Example

For example, for the Fourier basis stated above, if we also choose the variances ψ_m^2 appropriately:

$$\psi_m^2 = \frac{\eta^2}{(m-1)/2}$$

then it can be shown that

$$\lim_{M \to \infty} K(\mathbf{x}, \mathbf{x}') = \eta^2 \times \exp\left(-\rho^2(x - x')/2\right)$$

This gives us our radial basis function kernel from earlier in the course.

Basis Functions and Covariance Functions

To summarize:

- A Gaussian prior on β implies a Gaussian prior on $\mathbf{y}$
- This implicitly specifies a marginal likelihood of $\mathbf{y}$ that is Gaussian, and achieves what we set out to do in lecture 8 (adds dependencies into the regression model)
- This implies a joint Gaussian prior on the training and test data, which gives a predictive distribution for a test case y^* that is Gaussian.
- If we choose the basis functions in a clever way, we can get a convergent infinite sum that gives us a nice noise-free covariance function

So all we have to do is choose basis functions that give us a nice series.

Gaussian Process Models

Or...

Why even bother specifying basis functions? We could just specify the covariance function directly!

If the point of having a basis function model is to try and encode into the model facts about how we think $\mathbf{x}$ changes the predictive distribution of y, then specifying the covariance between y and y' as a function of $\mathbf{x}$ and $\mathbf{x}'$ makes sense.

Of course if you literally have a particular basis function expansion in mind, you could still just use it. The point is that you don't *have* to encode the relationship between training/test points in this specific way.

Gaussian Process Models

Consider the following model:

$$\begin{aligned} y_i &= f(\mathbf{x}_i) + \epsilon_i \\ \epsilon_i &\sim N(0, \sigma^2) \\ Cov(f(\mathbf{x}_i), f(\mathbf{x}_{i'})) &= K(\mathbf{x}_i, \mathbf{x}_{i'}) \end{aligned}$$

Now specify that for *any* set of features $\mathbf{x}_1, \ldots, \mathbf{x}_p$,

$$\left(f(\mathbf{x}_1), \ldots, f(\mathbf{x}_p)\right) \sim N\left(0, \mathbf{K}\right)$$

Gaussian Process Models

Definiton: a **Gaussian Process** is any collection of random variables, every finite subset of which has a joint Gaussian distribution.

This defines a joint distribution of any arbitrary set of function values $f(\mathbf{x}_i)$. There is a theorem from stochastic processes that says that this is sufficient to specify a prior distribution *over functions*, f.

f is what's random here, not $\mathbf{x}$. Hence a GP is a stochastic process defined by the characteristic that any set of realizations of the process have a joint Gaussian distribution. It is completely specified by the covariance function $K(\mathbf{x}, \mathbf{x}')$.

Covariance Functions

The problem of choosing basis functions then has been converted into the problem of choosing covariance functions.

We can use pretty much anything, as long as it produces a covariance matrix which is positive definite.

It is hard to check this for a given form of $K(\mathbf{x}, \mathbf{x}')$, but there are some known classes of covariance functions that do provably satisfy this requirement.

Covariance Functions

Consider, for the moment, a univariate feature x (more on this in a few slides).

We can look at a few examples of covariance functions, also known as **kernels**, that are known to be *valid*:

$$\begin{aligned} \text{Constant: } & K(x, x') = \gamma \\ \text{Linear: } & K(x, x') = \gamma x x' \end{aligned}$$

Covariance Functions

These aren't arbitrary: each kernel comes from a particular model.

The Constant kernel comes from $f(x) = \mu$, with $\mu \sim N(0, \gamma)$.

The Linear kernel comes from $f(x) = \beta x$, with $\beta \sim N(0, \gamma)$.

Covariance Functions

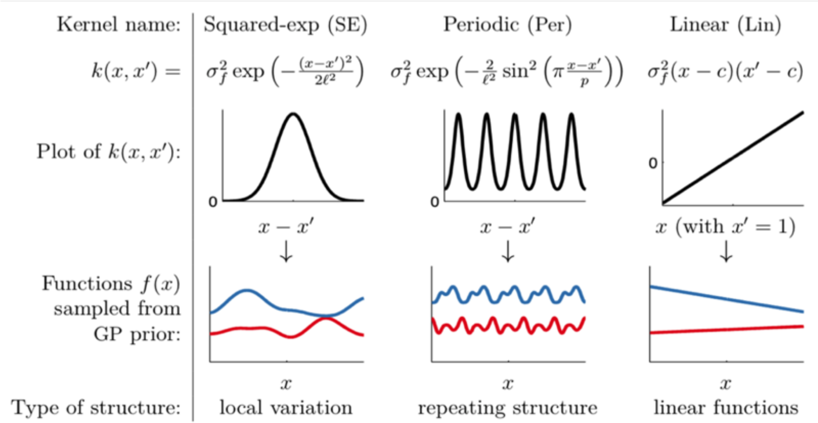

Figure 1: Source: David Duvenaud's PhD thesis, chapter 2, figure 1.1

Assumptions

Choosing a kernel is to choose a form of the random function, the distribution of which we are specifying a prior over.

One of the big classes of assumptions that we can (if we choose to) make is **stationarity**.

This is when we pick a kernel that is invariant to location shifts in x, i.e. one that depends only on $\|\mathbf{x} - \mathbf{x}'\|$:

$$K(\mathbf{x}, \mathbf{x'}) \equiv K\left(\left\|\mathbf{x} - \mathbf{x}'\right\|\right)$$

For example, the squared-error kernel

$$K(x, x') = \sigma^2 \exp\left(-\frac{1}{2\ell^2}(x - x')^2\right)$$

Multivariate Kernels

Choosing a kernel in one dimension is a fathomable problem. How do we extend this to multivariate features $\mathbf{x}$?

Key point: if k_1 and k_2 are valid kernels (produce valid covariance matrices) then

$$k = k_1 + k_2$$

and

$$k = k_1 k_2$$

are also valid kernels.

We can use this notion to build up multivariate kernels from univariate ones, and to combine univariate kernels in new ways.

Multivariate Kernels

For example, a fully linear kernel in p inputs with an intercept has an underlying function of the form

$$f(\mathbf{x}) = \sum_{j=1}^{p} f_j(x_j); f_j(x) = \beta_j x; \beta_j \sim N(0, \gamma_j)$$

which is obtained from adding linear kernels:

$$K(\mathbf{x}, \mathbf{x}') = \sum_{j=0}^{p} K_j(x_j, x_j')$$

where

$$K_0(x, x') = \gamma_0$$
$$K_j(x, x') = \gamma_j x x'$$

Combining Univariate Kernels

You could also combine multiple kernels on the same feature. For example, polynomials:

$$K_s(x, x') = \prod_{j=1}^{s} K_j(x, x')$$

where each K_j is a linear kernel, gives

$$K_s(x, x') = \gamma_1 (xx')^s$$

Adding these together for different s, for each feature, gives you a multivariate polynomial kernel.

Combining Univariate Kernels

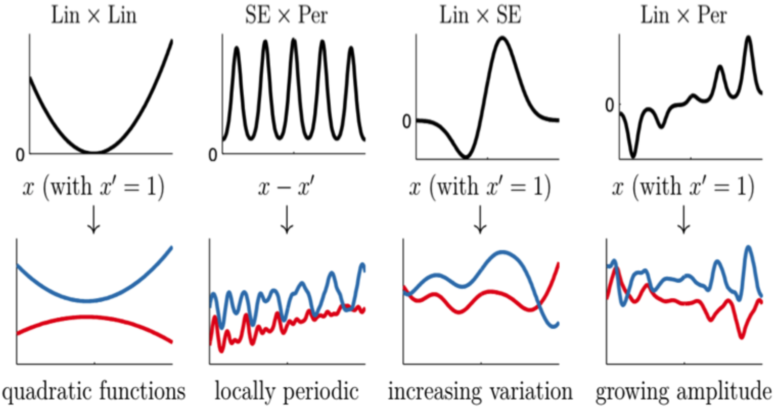

Figure 2: Source: David Duvenaud's PhD thesis, chapter 2, figure 1.2

Combining Univariate Kernels

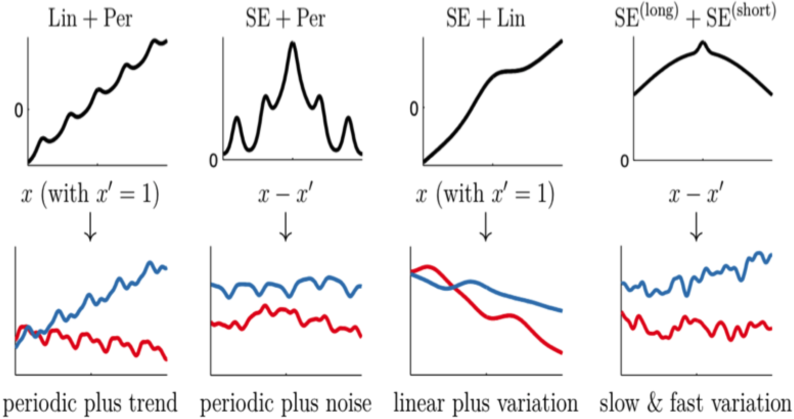

Figure 3: Source: David Duvenaud's PhD thesis, chapter 2, figure 1.4

Model Selection

Because inference is done by maximizing the likelihood, model selection is not that different from any other class of models.

As before, we can

- Compare (appropriately penalized) likelihoods for nested models
- Compare test set accuracy

Hyper-parameters of a particular kernel can be chosen by cross validation

Comments

Why is this different than choosing basis functions?

Choosing basis functions requires you to have an idea of the way y and $\mathbf{x}$ are related, and then guess at a function that induces that relation through its equivalent kernel.

Choosing a kernel directly is much more interpretable. And if you have a list of properties that you think are present between y and $\mathbf{x}$, you can directly combine kernels with these properties to get a new kernel with all the properties that you want.

E.g. if you think your data has periodic structure, use a periodic kernel. But maybe the fit is not too good because the amplitude grows with $\mathbf{x}$- combine with a linear kernel!

Further Reading

- Gaussian Processes:
  - Rasmussen and Williams (2006): *Gaussian Processes for Machine Learning*
- Choosing Kernels:
  - David Duvenaud's PhD thesis (especially chapters 2 and 3): https://raw.githubusercontent.com/duvenaud/phd-thesis/master/kernels.pdf
  - ... and his Kernel cookbook: https://www.cs.toronto.edu/∼duvenaud/cookbook/