Gaussian Processes

John Shawe-Taylor
Supervised Learning
jst@cs.ucl.ac.uk

November, 2017

Aim:

Today's lectures cover the links between kernel methods and Bayesian inference. This will include:

- Quick review of the Kernel methods approach
- Worked example of kernel Ridge Regression
- Properties of kernels.
- Bayesian inference and its application to regression
- Gaussian Process regression

Kernel methods

Kernel methods (re)introduced in 1990s with Support Vector Machines

- Linear functions but in high dimensional spaces equivalent to non-linear functions in the input space

- Statistical analysis showing large margin can overcome curse of dimensionality

- Extensions rapidly introduced for many other tasks other than classification

Kernel methods approach

- Data embedded into a Euclidean feature space
- Linear relations are sought among the images of the data
- Algorithms implemented so that only require inner products between vectors
- Embedding designed so that inner products of images of two points can be computed directly by an efficient ‘short-cut’ known as the kernel.

Worked example: Ridge Regression

Consider the problem of finding a homogeneous real-valued linear function

$$g(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle = \mathbf{x}'\mathbf{w} = \sum_{i=1}^{n} w_i x_i,$$

that best interpolates a given training set

$$S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$$

of points $\mathbf{x}_i$ from $X \subseteq \mathbb{R}^n$ with corresponding labels y_i in $Y \subseteq \mathbb{R}$.

Least squares regression

- Measures discrepancy between function output and correct output – squared to ensure always positive:
$$(g(\mathbf{x}) - y)^2$$

- We introduce notation: matrix $\mathbf{X}$ has rows the m examples of S. Hence we can write
$$\xi = \mathbf{y} - \mathbf{X}\mathbf{w}$$
for the vector of differences between $g(\mathbf{x}_i)$ and y_i.

Regularising to control flexibility of $\mathbf{w}$

Need to ensure flexibility of the regression is controlled – controlling the norm of $\mathbf{w}$ proves effective – this is known as regularisation:

$$\min_{\mathbf{w}} \mathcal{L}_\lambda(\mathbf{w}, S) = \min_{\mathbf{w}} \lambda \|\mathbf{w}\|^2 + \|\xi\|^2,$$

where we can compute

$$\begin{aligned} \|\xi\|^2 &= \langle \mathbf{y} - \mathbf{X}\mathbf{w}, \mathbf{y} - \mathbf{X}\mathbf{w} \rangle \\ &= \mathbf{y}'\mathbf{y} - 2\mathbf{w}'\mathbf{X}'\mathbf{y} + \mathbf{w}'\mathbf{X}'\mathbf{X}\mathbf{w} \end{aligned}$$

Setting derivative of $\mathcal{L}_\lambda(\mathbf{w}, S)$ equal to 0 gives

$$\mathbf{X}'\mathbf{X}\mathbf{w} + \lambda\mathbf{w} = \left(\mathbf{X}'\mathbf{X} + \lambda\mathbf{I}_n\right)\mathbf{w} = \mathbf{X}'\mathbf{y}$$

Primal solution

We get the primal solution weight vector:

$$\mathbf{w} = \left(\mathbf{X}'\mathbf{X} + \lambda \mathbf{I}_n\right)^{-1} \mathbf{X}'\mathbf{y}$$

and regression function

$$g(\mathbf{x}) = \mathbf{x}'\mathbf{w} = \mathbf{x}' \left(\mathbf{X}'\mathbf{X} + \lambda \mathbf{I}_n\right)^{-1} \mathbf{X}'\mathbf{y}$$

Dual solution

A dual solution should involve only computation of inner products – this is achieved by expressing the weight vector as a linear combination of the training examples:

$$\mathbf{X}'\mathbf{X}\mathbf{w} + \lambda\mathbf{w} = \mathbf{X}'\mathbf{y} \quad \text{implies}$$

$$\mathbf{w} = \frac{1}{\lambda}(\mathbf{X}'\mathbf{y} - \mathbf{X}'\mathbf{X}\mathbf{w}) = \mathbf{X}'\frac{1}{\lambda}(\mathbf{y} - \mathbf{X}\mathbf{w}) = \mathbf{X}'\alpha,$$

where

$$\alpha = \frac{1}{\lambda}(\mathbf{y} - \mathbf{X}\mathbf{w}) \tag{1}$$

or equivalently

$$\mathbf{w} = \sum_{i=1}^{m} \alpha_i \mathbf{x}_i$$

Dual solution

Substituting $\mathbf{w} = \mathbf{X}'\alpha$ into equation (1) we obtain:

$$\lambda\alpha = \mathbf{y} - \mathbf{X}\mathbf{X}'\alpha$$

implying

$$\left(\mathbf{X}\mathbf{X}' + \lambda\mathbf{I}_m\right)\alpha = \mathbf{y}$$

This gives the dual solution:

$$\alpha = \left(\mathbf{X}\mathbf{X}' + \lambda\mathbf{I}_m\right)^{-1}\mathbf{y}$$

and regression function

$$g(\mathbf{x}) = \mathbf{x}'\mathbf{w} = \mathbf{x}'\mathbf{X}'\alpha = \sum_{i=1}^{m} \alpha_i \langle \mathbf{x}, \mathbf{x}_i \rangle$$

Key ingredients of dual solution

Step 1: Compute

$$\alpha = (\mathbf{K} + \lambda \mathbf{I}_m)^{-1} \mathbf{y}$$

where $\mathbf{K} = \mathbf{X}\mathbf{X}'$ that is $\mathbf{K}_{ij} = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$

Step 2: Evaluate on new point $\mathbf{x}$ by

$$g(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i \langle \mathbf{x}, \mathbf{x}_i \rangle$$

Important observation: Both steps only involve inner products

Applying the ‘kernel trick’

Since the computation only involves inner products, we can substitute for all occurrences of $\langle \cdot, \cdot \rangle$ a kernel function κ that computes:

$$\kappa(\mathbf{x}, \mathbf{z}) = \langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle$$

and we obtain an algorithm for ridge regression in the feature space F defined by the mapping

$$\phi : \mathbf{x} \longmapsto \phi(\mathbf{x}) \in F$$

Note if ϕ is the identity this has no effect.

A simple kernel example

The simplest non-trivial kernel function is the quadratic kernel:

$$\kappa(\mathbf{x}, \mathbf{z}) = \langle \mathbf{x}, \mathbf{z} \rangle^2$$

involving just one extra operation. But surprisingly this kernel function now corresponds to a complex feature mapping:

$$\begin{aligned} \kappa(\mathbf{x}, \mathbf{z}) &= (\mathbf{x}'\mathbf{z})^2 = \mathbf{z}'(\mathbf{x}\mathbf{x}')\mathbf{z} \\ &= \langle \mathrm{vec}(\mathbf{z}\mathbf{z}'), \mathrm{vec}(\mathbf{x}\mathbf{x}') \rangle \end{aligned}$$

where $\mathrm{vec}(A)$ stacks the columns of the matrix A on top of each other. Hence, κ corresponds to the feature mapping

$$\phi : \mathbf{x} \longmapsto \mathrm{vec}(\mathbf{x}\mathbf{x}')$$

Implications of the kernel trick

- Consider for example computing a regression function over 1000 images represented by pixel vectors – say $32 \times 32 = 1024$.

- By using the quadratic kernel we implement the regression function in a $1,000,000$ dimensional space

- but actually using less computation for the learning phase than we did in the original space.

Implications of kernel algorithms

- Can perform linear regression in very high-dimensional (even infinite dimensional) spaces efficiently.

- This is equivalent to performing non-linear regression in the original input space: for example quadratic kernel leads to solution of the form
$$g(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle^2$$
that is a quadratic polynomial function of the components of the input vector $\mathbf{x}$.

- Using these high-dimensional spaces must surely come with a health warning, what about the curse of dimensionality?

Baysian Interpretation

- The Bayesian analysis of learning moves the source of randomness from the generation of the examples to two new sources:
  - A prior distribution over the possible choice of hypotheses
  - An additive noise model for inaccuracy in the observation of the outputs

- The randomness in the generation of the examples is ignored – basically the analysis is conditioned on the sample inputs.

- We first consider the two sources of randomness

Additive Noise model

- The most popular additive noise model is that of Gaussian noise.

$$P(y|\mathbf{w},\mathbf{x}) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(y-\langle \mathbf{w},\phi(\mathbf{x})\rangle)^2}{2\sigma^2}\right)$$

- The motivation for this choice is that the output of the linear function is the ‘true’ output

- This true value is then corrupted by additive noise with zero mean and variance σ^2 to give the observed training output $y \in \mathbb{R}$.

Prior distribution

- $\mathcal{F}$ is set of linear functions in F with density $dP(f)$ given by a symmetrical Gaussian distribution centred at the origin over the weights $\mathbf{w} \in F$, that is

  $\mathcal{F} = \{f(\mathbf{x}) = \langle \mathbf{w}, \phi(\mathbf{x}) \rangle : \mathbf{w} \in F\}$ and $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

  We can compute the covariance $C(\mathbf{x}, \mathbf{z})$ between the function outputs for two inputs $\mathbf{x}$ and $\mathbf{z}$:

$$\begin{aligned} C(\mathbf{x}, \mathbf{z}) &= \int_{\mathcal{F}} f(\mathbf{x}) f(\mathbf{z}) dP(f) \\ &= \int_{F} \phi(\mathbf{x})' \mathbf{w}\mathbf{w}' \phi(\mathbf{z}) d\mathcal{N}(\mathbf{w}) \\ &= \phi(\mathbf{x})' \int_{F} \mathbf{w}\mathbf{w}' d\mathcal{N}(\mathbf{w}) \phi(\mathbf{z}) \\ &= \phi(\mathbf{x})' \mathbf{I} \phi(\mathbf{z}) = \kappa(\mathbf{x}, \mathbf{z}) \end{aligned}$$

Bayesian Inference

- With a prior and additive noise model defined we can now describe what is meant by Bayesian inference – this develops the idea of likelihood by including the prior in the probability calculations.

- This produces a posterior distribution for the functions in $\mathcal{F}$:

$$dP_{\mathrm{post}}(f) = dP(f)P(\mathbf{y}|S_{\mathbf{x}}, f),$$

where $P(\mathbf{y}|S_{\mathbf{x}}, f)$ is calculated using the noise model ($S_{\mathbf{x}}$ denotes the training inputs).

Bayesian Inference

- Using the independence assumption

$$
\begin{aligned}
P(\mathbf{y}|S_{\mathbf{x}}, f) \quad &\propto \quad \prod_{i=1}^{m} \exp\left(-\frac{(y_i - f(\mathbf{x}_i))^2}{2\sigma^2}\right) \\
&= \quad \exp\left(-\frac{\sum_{i=1}^{m}(y_i - f(\mathbf{x}_i))^2}{2\sigma^2}\right).
\end{aligned}
$$

- combining with the prior gives a posterior distribution for the Gaussian Process case of

$$
dP(\mathbf{w})P(\mathbf{y}|S_{\mathbf{x}}, \mathbf{w}) \propto
$$

$$
\exp\left(-\frac{\sum_{i=1}^{m}(y_i - \langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle)^2}{2\sigma^2} - \frac{\|\mathbf{w}\|^2}{2}\right).
$$

Bayesian Inference

- The simplest inference is to use the map (maximum a posteriori probability) solution by simply choosing the $\mathbf{w}$ that maximises the posterior distribution. Using the monotonicity of the exponential function this gives

$$\mathbf{w}_{\mathrm{map}} = \mathrm{argmin}_{\mathbf{w}} \sum_{i=1}^{m} (y_i - \langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle)^2 + \sigma^2 \|\mathbf{w}\|^2$$

which is the Ridge Regression optimisation criterion with $\lambda = \sigma^2$, i.e.

$$\mathbf{w}_{\mathrm{map}} = \mathbf{X}'(\mathbf{C} + \sigma^2 \mathbf{I})^{-1} \mathbf{y}.$$

Dual solution

Substituting $\mathbf{w}_{\mathrm{map}} = \mathbf{X}'\alpha$ into this equation we obtain:

$$\sigma^2\alpha = \mathbf{y} - \mathbf{C}\alpha$$

implying

$$\left(\mathbf{C} + \sigma^2\mathbf{I}_m\right)\alpha = \mathbf{y}$$

This gives the dual solution:

$$\alpha = \left(\mathbf{X}\mathbf{X}' + \sigma^2\mathbf{I}_m\right)^{-1}\mathbf{y}$$

and regression function

$$g(\mathbf{x}) = \mathbf{x}'\mathbf{w} = \mathbf{x}'\mathbf{X}'\alpha = \sum_{i=1}^{m}\alpha_i C(\mathbf{x}, \mathbf{x}_i)$$

Regularisation versus prior distribution

- Note how the Gaussian prior distribution has exactly the same effect as the regularisation of the 2-norm of the weight vector $\mathbf{w}$: these are equivalent.

- Note also how the Gaussian noise model corresponds to minimising the squared discrepancy: again least squares corresponds to a Gaussian noise model.

- Hence, if we perform map inference, Gaussian process regression and Kernel Ridge Regression are equivalent.

Gaussian distributions

- Gaussian distributions are an example of the exponential family in which probability is proportional to

$$P(\mathbf{x}) \propto \exp(g(\mathbf{x}))$$

- The distribution is defined by choosing different $g(\cdot)$ and domain for $\mathbf{X}$. For most choices computing the normalisation constant is hard.

Gaussian distributions

- For the Gaussian g is chosen to be a quadratic function and the domain Euclidean space. In this case the distribution is well defined if g is negative definite

  $$g(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \mu)'\Sigma^{-1}(\mathbf{x} - \mu)$$

  for Σ positive definite. The normalisation constant is then

  $$\det(2\pi\Sigma)^{-1/2}$$

- The Covariance matrix of the Gaussian is of course Σ

Gaussian distributions

- If we apply a non-singular linear function $\mathbf{B}$ to the Euclidean space $\mathbf{z} = \mathbf{B}\mathbf{x}$ a Gaussian distribution is defined since the conditions continue to hold:

  $$f(\mathbf{z}) = g(\mathbf{B}^{-1}\mathbf{z}) = -\frac{1}{2}(\mathbf{B}^{-1}\mathbf{z} - \mu)'\Sigma^{-1}(\mathbf{B}^{-1}\mathbf{z} - \mu)$$

  with mean $\mathbf{B}\mu$ and covariance $\mathbf{B}\Sigma\mathbf{B}'$

- Intuitively this follows from the fact that a positive definite function defines a rugby ball and this remains true in a change of basis.

Prior distribution

- As a Gaussian distribution projected onto any subspace is still a Gaussian, if we consider a set of training examples

  $$S = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$$

  the distribution of corresponding output values

  $$y_1, \ldots, y_m$$

  will be distributed according to the multidimensional Gaussian with mean 0 and (positive definite) covariance matrix

  $$\mathbf{K} = (K_{ij} = C(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1}^{m}$$

Gaussian Process

- A point in the general Gaussian distribution corresponds to an assignment of outputs for every input – that is to a function $f(\mathbf{x}) \in \mathcal{F}$.

- Prior distribution can be viewed as a Gaussian with a dimension for each possible input whose value is the corresponding output!

- Such an infinite dimensional Gaussian distribution is known as a Gaussian process (GP)

- It is defined by a mean function (in our case $\mathbf{0}$) and the positive semi-definite covariance function $C(\mathbf{x}, \mathbf{z})$ of pairs of inputs.

- Formally, the definition requires that the distribution marginalised to any finite subset is a Gaussian with the restricted mean and covariance functions.

Gaussian Process Inference

- Using the map solution throws away information about the posterior distribution. Full Bayesian inference uses the average of the output over the posterior distribution as the output:

$$\mathbf{x} \mapsto \int_{\mathcal{F}} f(\mathbf{x}) dP_{\mathrm{post}}(f) = \int_{F} \langle \mathbf{w}, \phi(\mathbf{x}) \rangle dP_{\mathrm{post}}(\mathbf{w})$$

- Since the posterior is a Gaussian, the mean and the map solution coincide and since the mean of the output of a random linear function is the output of the mean function, the map and Bayesian average coincide.

Gaussian Process Inference

- But – we have more information than just the predicted output: we have the complete posterior distribution:

$$\begin{aligned} P_{\mathrm{post}}(\mathbf{w}|S) &\propto \exp\left(-\frac{\sum_{i=1}^{m}(y_i - \langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle)^2}{2\sigma^2} - \frac{\|\mathbf{w}\|^2}{2}\right) \\ &\propto \exp\left(-\frac{-2y_i'\mathbf{X}\mathbf{w} + \mathbf{w}'\mathbf{X}'\mathbf{X}\mathbf{w}}{2\sigma^2} - \frac{\|\mathbf{w}\|^2}{2}\right) \\ &\propto \exp\left(-\frac{1}{2}(\mathbf{w} - \mathbf{w}_{\mathrm{map}})'\mathbf{\Sigma}^{-1}(\mathbf{w} - \mathbf{w}_{\mathrm{map}})\right), \end{aligned}$$

where $\mathbf{w}_{\mathrm{map}}$ is the map solution and

$$\mathbf{\Sigma}^{-1} = \frac{1}{\sigma^2}(\mathbf{X}'\mathbf{X} + \sigma^2\mathbf{I})$$

i.e. a Gaussian with mean $\mathbf{w}_{\mathrm{map}}$ and covariance $\mathbf{\Sigma}$.

Error bars on the output

we can compute error bars on the outputs as we can work out the variance of the output that results from the posterior distribution of weights:

$$y = \mathbf{w}'_{\mathrm{map}}\phi(\mathbf{x}) + \mathbf{v}'\sqrt{\boldsymbol{\Sigma}}\phi(\mathbf{x}),$$

where $\mathbf{v}$ are independent zero mean unit variance Gaussian variables, so that

$$\begin{aligned}\sigma_y^2 = \mathrm{E}[(y - \mathbf{w}'_{\mathrm{map}}\phi(\mathbf{x}))^2] &= \phi(\mathbf{x})'\sqrt{\boldsymbol{\Sigma}}\mathbf{v}\mathbf{v}'\sqrt{\boldsymbol{\Sigma}}\phi(\mathbf{x}) \\ &= \phi(\mathbf{x})'\boldsymbol{\Sigma}\phi(\mathbf{x}) \\ &= \kappa(\mathbf{x}, \mathbf{x}) - \mathbf{k}'(\mathbf{K} + \sigma^2\mathbf{I})^{-1}\mathbf{k}.\end{aligned}$$

Bayesian inference: evidence

- We can now use the Bayesian inference to compute the 'evidence' for the model: this is the probability of the data:

$$P(\mathbf{y}) = \int_{\mathcal{F}} P_{\mathrm{post}}(f|S)df.$$

In the case we are considering this is just the integral of a Gaussian which can be estimated by observing that the posterior is a Gaussian, so that

$$\begin{aligned} \log(P(\mathbf{y}|X)) &= -\frac{1}{2}\mathbf{y}'(\mathbf{K}+\sigma^2\mathbf{I})^{-1}\mathbf{y} - \\ &\quad -\frac{1}{2}\log\det(\mathbf{K}+\sigma^2\mathbf{I}) - \frac{m}{2}\log 2\pi. \end{aligned}$$

Use of evidence

- Maximising this quantity can be used to perform model selection over different choices of the noise variance σ^2 or to set other hyperparameters such as the width of a Gaussian kernel.

Approximate Bayesian inference

- We have looked in some detail at Bayesian inference in the case of regression with a Gaussian noise model and Gaussian prior.

- This combination ensures that we can compute the posterior exactly.

- Different priors and noise models (eg for classification) will lead to different inference computations, which in general will not be solvable explicitly – this leads to the need for approximation techniques such as
  - Laplace approximation
  - variational inference