Nonlinear Regression: Gaussian Process

By: John Donaghy

Outline

- Linear Regression
  - Extension to nonlinear regression
- Gaussian Process
  - Bayes Rule
  - Update to linear regression
  - Transition to function space view

Linear Regression

$$\mathbf{y} = \theta^T \mathbf{x} + \epsilon \tag{3}$$

define residual as $r \equiv \mathbf{y} - \theta^T \mathbf{x} = \epsilon$, as $n \to \infty$ allows us to use the central limit theorem

$$\epsilon \sim \mathcal{N}(0, 1)$$

$$y_i \sim \mathcal{N}(\theta^T \mathbf{x}_i, \sigma^2)$$

Maximum Likelihood: assume that your samples were drawn from the most probable distribution

$$p(\mathbf{y}|\mathbf{x}) = \prod_i p(y_i|\mathbf{x}_i)$$

$$p(y_i|\mathbf{x}_i) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{(y_i - \theta^T \mathbf{x}_i)^2}{2\sigma^2}}$$

Linear Regression

$$\mathbf{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2^2 + \cdots \tag{1}$$

$$\mathbf{y} = \theta_0 + \theta_1 f_1(x_1) + \theta_2 f_2(x_2) + \cdots$$

Gaussian Process

- Posterior distribution over all functions
- Incorporate expert (domain) knowledge
  - Great for scientists!
- Interpolates observations with empirical confidence intervals
- “Knows what it does not know”

Bayes Rule

$$p(b|a) = \frac{p(a|b)p(b)}{p(a)} \tag{2}$$

$p(b|a)$: posterior
$p(a|b)$: likelihood
$p(b)$: prior
$p(a)$: marginal likelihood

Bayes rule allows us to update a probability distribution by observing training data.
For the following we will drop the marginal likelihood as it is a normalization constant
And assume gaussian normal distribution functions.

Bayes Rule to predict function

From linear regression,

$$\epsilon \sim \mathcal{N}(0, \sigma)$$

which allows us to say

$$p(\mathbf{y}|\mathbf{X}, \theta) \sim \mathcal{N}(\mathbf{X}^T\theta, \sigma^2\mathbf{I})$$

If we begin by assuming a gaussian prior over the parameters

$$p(\theta) \sim \mathcal{N}(0, \sigma_p)$$

we can apply Bayes to obtain the posterior distribution

$$p(\theta|\mathbf{y}, \mathbf{X}) = p(\mathbf{y}|\mathbf{X}, \theta)p(\theta)$$

$$p(y_*|\mathbf{X}, \mathbf{y}, \mathbf{x}_*) = \int p(y_*|\mathbf{x}_*, \theta)p(\theta|\mathbf{y}, \mathbf{X})d\theta = \mathcal{N}(\mu, \Sigma)$$

- A GP is completely specified by a mean function and a covariance function
- Instead of integrating over all weights, introduce a kernel function for covariance matrix
- Projects to high dimension feature space. Source of nonlinearity.
- Computationally more efficient

Multivariate Joint Normals

$$\mathbf{y} \sim \frac{1}{\sqrt{(2\pi)^k |\boldsymbol{\Sigma}|}} e^{\left(-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{y}-\boldsymbol{\mu})\right)} \tag{4}$$

This is what is known as a joint normal distribution. We can extend the joint normal to include our inference points (*) **"Update our Prior"**

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{y}_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(X,X) + \sigma^2\mathbf{I} & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix}\right) \tag{5}$$

Using (a lot of) math we can construct a conditional distribution for our inference point

$$\mathbf{y}_* | X, y, X_* \sim \mathcal{N}(\mu_*, \Sigma)$$

$$\mu_* = K(X_*, X)\left(K(X,X) + \sigma^2\mathbf{I}\right)^{-1}\mathbf{y}$$

$$\Sigma = K(X_*,X_*) - K(X_*,X)\left(K(X,X) + \sigma^2\mathbf{I}\right)^{-1}K(X,X_*)$$

Sample from MVG

Computational methods[edit]

Drawing values from the distribution[edit]

A widely used method for drawing (sampling) a random vector **x** from the *N*-dimensional multivariate normal distribution with mean vector **μ** and covariance matrix **Σ** works as follows:[33]

1. Find any real matrix **A** such that $\mathbf{A}\,\mathbf{A}^{\mathrm{T}} = \mathbf{\Sigma}$. When **Σ** is positive-definite, the Cholesky decomposition is typically used, and the extended form of this decomposition can always be used (as the covariance matrix may be only positive semi-definite) in both cases a suitable matrix **A** is obtained. An alternative is to use the matrix $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}^{1/2}$ obtained from a spectral decomposition $\mathbf{\Sigma} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$ of **Σ**. The former approach is more computationally straightforward but the matrices **A** change for different orderings of the elements of the random vector, while the latter approach gives matrices that are related by simple re-orderings. In theory both approaches give equally good ways of determining a suitable matrix **A**, but there are differences in computation time.
2. Let $\mathbf{z} = (z_1, \ldots, z_N)^{\mathrm{T}}$ be a vector whose components are *N* independent standard normal variates (which can be generated, for example, by using the Box–Muller transform).
3. Let **x** be **μ** + **Az**. This has the desired distribution due to the affine transformation property.

https://en.wikipedia.org/wiki/Multivariate_normal_distribution#Drawing_values_from_the_distribution

Drawbacks of GP

- Requires large amount of memory: entire covariance matrix must be used for each inference
- Cholesky decomposition for inverse covariance matrix: O(n^3)