Kernel Methods

Ri Piao

Dept of IE, Seoul National University

March 3, 2017

Bishop, C. M. Pattern Recognition and Machine Learning *Information Science and Statistics*, Springer, 2006.

- Linear Regression Model whose parameters are determined by minimizing a regularized sum-of-squares error function:

$$J(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}(\mathbf{w}^\top \Phi(x_n) - t_n) + \frac{\lambda}{2}\mathbf{w}^\top \mathbf{w} \tag{1}$$

where $\lambda > 0$

$$\mathbf{w} = \frac{1}{\lambda}\sum_{n=1}^{N}(\mathbf{w}^\top \mathbf{\Phi}(\mathbf{x_n}) - \mathbf{t_n})\mathbf{\Phi}(\mathbf{x_n}) = \mathbf{\Phi}^\top \mathbf{a} \tag{2}$$

- Dual Representation:

$$J(\mathbf{w}) = \frac{1}{2}\mathbf{a}^\top \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{a} \tag{3}$$

- Gram matrix: $\mathrm{K}_n m = \phi(x_n)^\top \phi(x_m) = k(x_n, x_m)$
- Thus, equation(1) becomes:

$$J(\mathbf{a}) = \frac{1}{2}\mathbf{a}^\top \mathbf{K}\mathbf{K}\mathbf{a} - \mathbf{a}^\top \mathbf{K}\mathbf{t} + \frac{1}{2}\mathbf{t}^\top \mathbf{t} + \frac{\lambda}{2}\mathbf{a}^\top \mathbf{K}\mathbf{a} \tag{4}$$

Setting the gradient pf $J(\mathbf{a})$ with respect to $\mathbf{a}$ to zero, the solution is obtained as $\mathbf{a} = (\mathbf{K} + \lambda \mathbf{I}_N)^{-1}\mathbf{t}$
Subtitute back into the linear regression model:

$$y(\mathbf{x}) = \mathbf{w}^\top \phi((x)) = \mathbf{a}^\top \mathbf{\Phi}\phi(x) = \mathbf{k}(\mathbf{x})^\top (\mathbf{K} + I_N)^{-1}\mathbf{t} \tag{5}$$

- First approach: choose a feature space mapping $\phi(x)$ and then use this to find the corresponding kernel.
- Second approach: construct kernel directly.
  Ex) Consider a kernel function given by

$$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z})^2 \tag{6}$$

  Take the particular case of a two-dimensional input space $\mathbf{x} = (x_1, x_2)$, then k(x,z)=$\phi(\mathbf{x})^\top \phi(\mathbf{z})$
- A neccessary and sufficient condition for a function $k(\mathbf{x}, \mathbf{x}')$ to be a valid kernel is that Gram matrix $\mathbf{K}$, should be positive semidefinite for all possible choices of the set $\mathbf{x}_n$.
- "Gaussian" kernel: $k(\mathbf{x}, \mathbf{x}') = exp(-\|\mathbf{x} - \mathbf{x}\|^2/2\sigma^2)$
  It is not restricted to Euclidean spaces. It can be defined over objects as diverse as graphs, sets, strings, and text documents.
- Probabilistic generative model: $k(\mathbf{x}, \mathbf{x}') = p(\mathbf{x})p(\mathbf{x}')$
  With positive weighting coefficients: $k(\mathbf{x}, \mathbf{x}') = \sum_i p(\mathbf{x}|i)p(\mathbf{x}'|i)p(i)$
- Fisher kernel: $k(\mathbf{x}, \mathbf{x}') = \mathbf{g}(\theta, \mathbf{x})^\top \mathbf{F}^{-1}\mathbf{g}(\theta, \mathbf{x}')$ where $\mathbf{g}(\theta, \mathbf{x}) = \nabla_\theta lnp(\mathbf{x}|\theta)$
  We can just omit the Fisher information matrix, because it is often infeasible to evaluate the Fisher information matrix.

- Exact function interpolation:
  Goal: Given a set of input vectors $x_1, ..., x_N$ along with corresponding target values $t_1, ..., t_N$, then find a smooth function $f(x)$ that fits every target value exactly, so that $f(x_n) = t_n$ for $n = 1, ..., N$. Express $f(x)$ as a linear combination of radial basis functions: $f(x) = \sum_{n=1}^{N} w_n h(\|\mathbf{x} - \mathbf{x_n}\|)$.
  In pattern recognition applications, however. the target values are generally noisy, and exact interpolation is undesirable because this corresponds to an over-fitted solution.
- Another motivation: when inpit variables are noisy.
  Then the sum-of-squares error function becomes
  $E = \frac{1}{2} \sum_{n=1}^{N} \int \{y(\mathbf{x}_n + \xi) - t_n\}^2 \nu(\xi) d\xi$
  Using the caculus of variation, $y(\mathbf{x}_n) = \sum_{n=1}^{N} t_n h(x - x_n)$ where
  $h(x - x_n) = \frac{\nu(\mathbf{x} - \mathbf{x_n})}{\sum_{n=1}^{N} \nu(\mathbf{x} - \mathbf{x_n})}$
- Typically, the number of basis functions, and the locations $mathbf\, \mu_i$ of their centres, are determined based on the input data x_n alone.
- One of the simplest way of choosing basis function centres is to use a randomly chosen subset of the data points. A more systematic approach is called orthogonal least squares.

- Parzen density estimator to model the joint distribution $p(\mathbf{x}, t)$, so that

$$p(\mathbf{x}, t) = \frac{1}{N} \sum_{n=1}^{N} f(\mathbf{x} - \mathbf{x_n}, t - t_n) \tag{7}$$

where $f(\mathbf{x}, t)$ is the component density function. // Then

$$y(\mathbf{x}) = \mathbf{E}[t|\mathbf{x}] = \int_{-\infty}^{\infty} tp(t|\mathbf{x})dt = \frac{\int tp(\mathbf{x}, t)dt}{\int p(\mathbf{x}, t)dt} = \frac{\sum_n \int tf(\mathbf{x} - \mathbf{x_n}, t - t_n)}{\sum_m \int tf(\mathbf{x} - \mathbf{x_m}, t - t_m)} \tag{8}$$

Assume for simplicity that the component density functions have zero mean so that $\int_{-\infty}^{\infty} f(\mathbf{x}, t)tdt = 0$ for all values of x. Then

$$y(\mathbf{x}) = \frac{\sum_n g(\mathbf{x} - \mathbf{x_n})t_n}{\sum_m g(\mathbf{x} - \mathbf{x_m})t_m} = \sum_n k(\mathbf{x}, \mathbf{x}`_n)t_n \tag{9}$$

where $g(\mathbf{x}) = \int_{-\infty}^{\infty} f(\mathbf{x}, t)dt$

- Consider a model defined in terms of a linear combination of M fixed basis functions given by the elements of the vertor $\phi(\mathbf{x})$ so that $y(\mathbf{x}) = \mathbf{w}^\top \phi(\mathbf{x})$ If a prior distribution over $\mathbf{w}$ given by an isotropic Gaussian of the form: $p(\mathbf{w}) = N(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I})$

$$\mathbf{y} = \mathbf{\Phi w} \tag{10}$$

- Its mean and covariance are given by $\mathbf{E}[\mathbf{y}] = \mathbf{\Phi E}[\mathbf{w}] = 0$ and $cov[\mathbf{y}] = \mathbf{E}[\mathbf{y}\mathbf{y}^\top] = \mathbf{\Phi E}[\mathbf{w}\mathbf{w}^\top]\mathbf{\Phi}^\top = \frac{1}{\alpha}\mathbf{\Phi}\mathbf{\Phi}^\top = \mathbf{K}$
- A key point about Gaussian stochastic process is that the joint distribution over N variables $y_1, ...y_N$ is specified completely by the second-order statistics, namerly the mean and the covariance.
- We can also define the kernel function directly. Exponential kernel: $k(x, x') = exp(-\theta|x - x'|)$

- $t_n = y_n + \epsilon_n$ where $y_n = y(\mathbf{x}_n)$, and ϵ_n is a random noise variable. We consider noise processes that have a Gaussian distribution, so that $p(t_n|y_n) = N(t_n|y_n, \beta^{-1})$ where β is a hyperparameter representing the precision of the noise.
- The marginal distribution $p(\mathbf{y})$ is given by a Gaussian whose mean is zero and whose covariance is defined by a Gram matrix $\mathbf{K}$ so that $p(\mathbf{y}) = N(\mathbf{y}|\mathbf{0}, \mathbf{K})$
- The marginal distribution of $\mathbf{t}$ is given by $p(\mathbf{t}) = \int p(\mathbf{t}|\mathbf{y})p(\mathbf{y})d\mathbf{y} = N(\mathbf{t}|\mathbf{0}, \mathbf{C})$ where the covariance matrix $\mathbf{C}$ has elements $C(x_n, x_m) = k(x_n, x_m) + \beta^{-1}\delta_{nm}$. The distribution over $t_1, \ldots t_N + 1$ is given by $p(\mathbf{t}_{N+1}) = N(\mathbf{t}_{N+1}|\mathbf{0}, \mathbf{C}_{N+1})$ where
$$C_{N+1} = \begin{pmatrix} C_N & \mathbf{k} \\ \mathbf{k}^\top & c \end{pmatrix}$$
- The conditional distribution $p(t_N + 1|\mathbf{t})$ is a Gaussian distribution with mean and covariance given by

$$m(\mathbf{x}_{N+1}) = \mathbf{k}^\top C_N^{-1}\mathbf{t} \tag{11}$$

$$\sigma^2(\mathbf{x}_{N+1}) = c - \mathbf{k}^\top \mathbf{C}_N^{-1}\mathbf{k} \tag{12}$$

- The only restriction on the kernel function is that the covariance matric given by $C(x_n, x_m) = k(x_n, x_m) + \beta^{-1}\delta_{nm}$ must be positive definite.
- If λ_i is an eigenvalue of $\mathbf{K}$, then the corresponding eigenvalue of $\mathbf{C}$ will be $\lambda_i + \beta^{-1}$. It is therefore sufficient that the kernel matrix $k(\mathbf{x}_n, \mathbf{x}_m)$ be positive semidefinite for any pair of points $\mathbf{x}_n$ and $\mathbf{x}_m$, so that $\lambda_i >= 0$, because any eigenvalue λ_i that is zero will still give rise to a positive eigenvalue for $\mathbf{C}$ because $\beta > 0$.
- Advantage of Gaussian processes: we can consider covariance functions that can only be expressed in terms of an infinite number of basis functions.

- Techniques for learning the hyperparameters are based on the evaluation of the likelihood function $p(\mathbf{t}|\theta)$ where θ denotes the hyperparameters of the Gaussian process model.
- The log likelihood function for a Gaussian process regression model:

$$lnp(\mathbf{t}|\theta) = -\frac{1}{2}ln|\mathbf{C}_N| - \frac{1}{2}\mathbf{t}^\top \mathbf{C}_N^{-1}\mathbf{t} - \frac{N}{2}ln(2\pi) \tag{13}$$

- The derivative of $\frac{\partial \mathbf{C}_N}{\partial \theta_i}$ is given by

$$\frac{\partial}{\partial \theta_i} lnp(\mathbf{t}|\theta) = -\frac{1}{2} Tr(\mathbf{C}_N^{-1}\frac{\partial \mathbf{C}_N}{\partial \theta_i}) + \frac{1}{2}\mathbf{t}^\top \frac{\partial \mathbf{C}_N}{\partial \theta_i}\mathbf{C}_N^{-1}\mathbf{t} \tag{14}$$

- It have multiple maxima, because $lnp(\mathbf{t}|\theta)$ is nonconvex.

- Gaussian process with a two-dimensional input space $\mathbf{x} = (x_1, x_2)$, having a kernel function of the form

$$k(\mathbf{x}, \mathbf{x}') = \theta_0 exp \frac{1}{2} \sum_{i=1}^{2} \eta_i (x_i - x_i')^2 \tag{15}$$

- ARD framework is easily incorporated into the exponential-quadratic kernel to give the following form of kernel function

$$k(\mathbf{x_n}, \mathbf{x_m}) = \theta_0 exp \frac{1}{2} \sum_{i=1}^{D} \eta_i (x_{ni} - x_{mi})^2 + \theta_2 + \theta_3 \sum_{i=1}^{D} x_{ni} x_{mi} \tag{16}$$

where D is the dimensionality of the input space.

- Goal of probabilistic approach to classification: model the posterior probabilities of the target variable for a new input vector, given a set of training data.
- Denote the training set inputs by $\mathbf{x}_1, ..., \mathbf{x}_N$ with corresponding observed target variables $\mathbf{t} = (t_1, ..., t_N)^\top$.
- Goal: determine the predictive distribution $p(t_{N+1}|\mathbf{t})$.
  $\rightarrow$ introduce a Gaussian process prior over the vector $\mathbf{a}_{N+1} = N(\mathbf{a}_{N+1}|\mathbf{0}, \mathbf{C}_{N+1})$.
- The covariance matrix: $C(\mathbf{x}_n, \mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m) + \nu\delta_{nm}$

- Seek a Gaussian approximation to the posterior distribution over a_{N+1} given by

$$p(a_{N+1}|\mathbf{t}_N) = \int p(a_{N+1}, \mathbf{a}_N|\mathbf{t}_N)d\mathbf{a}_N = \frac{1}{p(\mathbf{t}_N)} \int p(a_{N+1}, \mathbf{a}_N|\mathbf{t}_N)p(\mathbf{t}_N|a_{N+1}, \mathbf{a}_N)d\mathbf{a}_N \tag{17}$$

- The prior is given by a zero-mean Gaussian process with covariance matrix $\mathbf{C}_N$, and the data term is given by $p(\mathbf{t}_N|\mathbf{a}_N) = \prod_{n=1}^{N} \sigma(a_n)^(t_n)(1-\sigma(a_n))^(1-t_n) = \prod_{n=1}^{N} e^{a_n t_n}\sigma(-a_n)$
- Additive normalization constant: $\Psi(\mathbf{a}_N) = lnp(N) + lnp(\mathbf{t}_N|\mathbf{a}_N) = -\frac{1}{2}\mathbf{a}_N^t\mathbf{C}_N^{-1}\mathbf{a}_N - \frac{N}{2}ln(2\pi) - \frac{1}{2}ln|\mathbf{C}_N| + \mathbf{t}_N^\top\mathbf{a}_N - \sum_{n=1}^{N} ln(1+e^{a_n}) + const.$
- The gradient of $\Psi(\mathbf{a}_N)$ is $\nabla\Psi(\mathbf{a}_N) = \mathbf{t}_N - \sigma_N - \mathbf{C}^{-1}\mathbf{a}_N$ and the second derivative of $\Psi(\mathbf{a}_N)$ is given by $\nabla\nabla\Psi(\mathbf{a}_N) = -\mathbf{W}_N - \mathbf{C}_N^{-1}$
- Using the Newton-Raphson formula:

$$\mathbf{a}_N^{new} = \mathbf{C}_N(\mathbf{I} + \mathbf{W}_N\mathbf{C}_N)^{-1}\mathbf{t}_N - \sigma_N + \mathbf{W}_N\mathbf{a}_N \tag{18}$$

  The equations are iterated until they converge to the mode: $\mathbf{a}_N^* = \mathbf{C}_N(\mathbf{t}_N - \sigma_N)$
- Evaluate the Hessian: $\mathbf{H} = -\nabla\nabla\Psi(\mathbf{a}_N)$

- Gaussian approximation to the posterior distribution $p(\mathbf{a}_N|\mathbf{t}_N)$ given by $q(\mathbf{a}_N) = N(\mathbf{a}_N|\mathbf{a}_N^*, \mathbf{H}^{-1})$

$$\mathbf{E}[a_{N+1}|\mathbf{t}_N] = \mathbf{k}^\top(\mathbf{t}_N - \sigma_N) \tag{19}$$

$$var[a_{N+1}|\mathbf{t}_N] = c - \mathbf{k}^\top(\mathbf{W}_N^{-1} + \mathbf{C}_N)^{-1}\mathbf{k} \tag{20}$$

- Determine the parameters θ of the covariance function.
  $\rightarrow$ Maximize the likelihood function: $p(\mathbf{t}_N|\theta) = \int p(\mathbf{t}_N|\mathbf{a}_N)p(\mathbf{a}_N|\theta)d\mathbf{a}_N$
  log of the likelihood function:
  $lnp(\mathbf{t}_N|\theta) = \Psi(\mathbf{a}_N^*) - \frac{1}{2}ln|\mathbf{W}_N + \mathbf{C}_{-1N}| + \frac{N}{2}ln(2\pi)$
  $\frac{\partial}{\partial\theta_i}lnp(\mathbf{t}_N|\theta) = \frac{1}{2}\mathbf{a}_N^*\mathbf{C}_N^{-1}\frac{\partial\mathbf{C}_N}{\partial\theta_j}\mathbf{C}_N^{-1}\mathbf{a}_N^* - \frac{1}{2}Tr[(\mathbf{I} + \mathbf{C}_N^{-1}\mathbf{W}_N)^{-1}\mathbf{W}_N\frac{\partial\mathbf{C}_N}{\partial\theta_j}$

$$-\frac{1}{2}\sum_{n=1}^{N}\frac{\partial ln|\mathbf{W}_N + \mathbf{C}_N^{-1}}{\partial a_n^*}\frac{\partial a_n^*}{\partial\theta_j} = -\frac{1}{2}\sum_{n=1}^{N}[(\mathbf{I}+\mathbf{C}_N\mathbf{W}_N)^{-1}\mathbf{C}_N]_{nn}\sigma_n^*(1-\sigma_n^*)(1-2\sigma_n^*)\frac{\partial a_n^*}{\partial\theta_j} \tag{21}$$

  Derivative of a_N^* with respect to θ_j is given by
  $\frac{\partial a_n^*}{\partial\theta_j} = \frac{\mathbf{C}_N}{\partial\theta_j}(\mathbf{t}_N - \sigma_N - \mathbf{C}_N\mathbf{W}_N\frac{\partial a_N^*}{\partial\theta_j})$
  Rearranging then gives: θ_j is given by $\frac{\partial a_n^*}{\partial\theta_j} = (\mathbf{I} + \mathbf{W}_N\mathbf{C}_N)^{-1}\frac{\partial\mathbf{C}_N}{\partial\theta_j}(\mathbf{t}_N - \sigma_N)$