HOLON INSTITUTE OF TECHNOLOGY, DEPARTMENT OF APPLIED MATHEMATICS

Sequential Models in Data Science
Bayesian Regression

Yirmeyahu Kaminski

INTRODUCTION

This exercise aims at giving an overall understanding of Bayesian Regression (also called Relevance Vector Machine) and to implement it in Python.
The exercise is to be done by pairs of students. Each pair must present a pdf file with the theoretical answers and a Jupyter notebook with the Python functions. The notebook will be tested as this. No modification should be necessary for making it work.

1 GENERAL REGRESSION PROBLEM

We are given n sampling times $x_1, \cdots, x_n$ and n corresponding targets $y_1, \cdots, y_n$. For each i, we define a radial function: $\phi_i(x) = \exp\left(-\frac{(x-x_i)^2}{r^2}\right)$.
The regression model is defined as the function $y(x,w) = \sum_{i=1}^{n} w_i \phi_i(x)$, where $w = (w_1, \cdots, w_n)$ is a vector of weights to be determined.
We assume that the targets are related to the model by an additive noise ϵ_i as follows:

$$y_i = y(x_i, w) + \epsilon_i.$$

The noise samples are assumed to be independent and identically distributed as $\mathcal{N}(0, \sigma^2)$.

2 COMPUTE THE POSTERIOR OF w

1. Write the likelihood of w.

2. We define the same prior for all weights: $\mathcal{N}(0, \alpha^{-1})$, where $\alpha > 0$. Write the Bayes formula that defines the posterior distribution of w.

3. Prove that computing the MAP estimate of w is equivalent to a regularized least square problem. Write the relation that relates α, σ^2 and λ, that is the regularization parameter.

3 The Gaussian Nature of the Posterior and the Prediction

In this section, we shall consider the two following lemmas. Note that you are not required to prove them.

Lemma 1. *If random variables $x \in \mathbb{R}^n$ and $y \in R^p$ have the Gaussian probability distributions:*

$$\begin{aligned} x &\sim \mathcal{N}(m, P) \\ y \mid x &\sim \mathcal{N}(Hx + u, R) \end{aligned}$$

then the joint distribution of (x, y) and the marginal distribution of y are given as:

$$\begin{aligned} \begin{pmatrix} x \\ y \end{pmatrix} &\sim \mathcal{N}\left(\begin{pmatrix} m \\ Hm + u \end{pmatrix}, \begin{pmatrix} P & PH^T \\ HP & HPH^T + R \end{pmatrix}\right) \\ y &\sim \mathcal{N}(Hm + u, HPH^T + R) \end{aligned}$$

Lemma 2. *If random variables $x \in \mathbb{R}^n$ and $y \in R^m$ have joint the Gaussian probability distributions:*

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} a \\ b \end{pmatrix}, \begin{pmatrix} A & C \\ C^T & B \end{pmatrix}\right)$$

then the marginal and conditional distribution of x and y are given as follows:

$$\begin{aligned} x &\sim \mathcal{N}(a, A) \\ y &\sim \mathcal{N}(b, B) \\ x \mid y &\sim \mathcal{N}(a + CB^{-1}(y - b), A - CB^{-1}C^T) \\ y \mid x &\sim \mathcal{N}(a + C^T A^{-1}(x - a), B - C^T A^{-1} C) \end{aligned}$$

Assuming that $w = (w_1, \ldots, w_n)^T$, $y = (y_1, \ldots, y_n)$ and the matrix Φ is the same as defined in lectures, answer the two following questions.

1. **Relying on these lemmas**, prove that the posterior distribution of w, i.e. $f(w \mid y, \alpha, \sigma^2)$ is $\mathcal{N}(\mu, \Sigma)$, where: $\Sigma^{-1} = \frac{1}{\sigma^2}\Phi^T\Phi + \alpha I_n$ and $\mu = \frac{1}{\sigma^2}\Sigma\Phi^T y$.

2. **Relying on these lemmas**, prove that the prediction distribution of $y_\star$ for a new point $x_\star$, i.e $f(y_\star \mid y) = \int f(y_\star \mid w, \hat{\sigma}^2, y)\pi(w \mid \hat{\alpha}) dw$ is $\mathcal{N}(\mu_\star, \sigma_\star)$, where: $\sigma_\star = \hat{\sigma}^2 + f^T \Sigma f$ and $\mu_\star = y(x_\star, \mu)$, where μ and Σ are defined by $f(w \mid y, \hat{\sigma}^2, \hat{\alpha}) = \mathcal{N}(\mu, \Sigma)$ and $f = [\phi_1(x_\star), \ldots, \phi_n(x_\star)]^T$. Here $\hat{\sigma}^2$ $\hat{\alpha}$ are the values of σ^2 and α given by type II maximum likelihood.

4 Sparse Bayesian Learning

The prior of each component w_i of w is now set separately $w_i \sim \mathcal{N}(0, \alpha_i^{-1})$, where $\alpha_i > 0$. The different components are assumed to be independent. Let $\alpha = (\alpha_1, \cdots, \alpha_n)$.
The posterior of the whole set of parameters is given by:

$$f(w, \alpha, \sigma^2 \mid y) = f(w \mid y, \alpha, \sigma^2)\pi(\alpha, \sigma^2).$$

We shall write from now β for $1/\sigma^2$ in order to make the equations less cluttered.

1. Show that $f(w \mid y, \alpha, \beta) = \mathcal{N}(m, \Sigma)$, where:

   $$m = \beta \Sigma \Phi^t y \text{ and } \Sigma = (A + \beta \Phi^t \Phi)^{-1},$$

   with: $A = \text{diag}(\alpha)$ and the matrix Φ is the same matrix as before.

2. We want to compute the parameters α and β using the Maximum Likelihood principle. For that purpose show that the log-likelihood $f(y \mid \alpha, \beta)$ is given by:

   $$\ln f(y \mid \alpha, \beta) = \frac{n}{2} \ln \beta - E(y) - \frac{1}{2} \ln |\Sigma| - \frac{n}{2} \ln(2\pi) + \frac{1}{2} \Sigma_{i=1}^{n} \ln \alpha_i,$$

   where $E(y) = \frac{1}{2}(\beta y^t y - m^t \Sigma^{-1} m)$. Once the log-likelihhod is given, one can design an iterative algorithm which estimates all the parameters.

5 Relevance Vector Machine (RVM)

The RVM process is an iterative one that implements Bayesian regression with sparsity, obtained by pruning point with big α_i. It involves repeatedly re-estimating $\alpha = (\alpha_1, \cdots, \alpha_n)^T$ and $\beta = \frac{1}{\sigma^2}$ until a stopping condition is met. In order to implement the algorithm, we must first:

1. Select a suitable kernel function, i.e. a suitable radial function, for the data set and relevant parameters. Use this kernel function to create the design matrix Φ.

2. Establish a suitable convergence criteria for α and β, e.g. a threshold value δ_{Thresh} for the change between one iteration estimation of α and the next, i.e. $\delta = \sum_{i=1}^{n} \mid \alpha_i^{k+1} - \alpha_i^k \mid$ so that re-estimation will stop when $\delta < \delta_{Thresh}$.

3. Establish a threshold value α_{Thresh} which it is assumed an α_i is tending to infinity upon reaching it.

4. Choose starting values for α and β.

Then the algorithm itself is described by the following steps:

`While` $\delta > \delta_{Thresh}$, `do:`

1. Calculate $m = \beta \Sigma \Phi^t y$ and $\Sigma = (A + \beta \Phi^t \Phi)^{-1}$, where $A = \text{diag}(\alpha)$.
2. Update $\alpha_i = \frac{\gamma_i}{m_i^2}$, where γ_i is computed relying on the previous values of α and Σ by the following expression: $\gamma_i = 1 - \alpha_i \Sigma_{ii}$.
3. Update $\beta = \frac{N - \sum_i \gamma_i}{\|y - \Phi m\|^2}$
4. Update δ
5. Prune the α_i and corresponding basis functions where $\alpha_i > \alpha_{Thresh}$.

Our hyperparameter values α and β which result from the above procedure are those that maximize our marginal likelihood and hence are those used when making a new estimate of a target value $y_\star$ for a new input $x_\star$:

$$y_\star = m^t \phi(x_\star),$$

with $\phi(x_\star) = [\phi_{i_1}(x_\star), \ldots, \phi_{i_p}(x_\star)]$ where p is the number of remaining basis functions.
The variance relating to our confidence in this estimate is given by:

$$\sigma_\star^2 = \beta^{-1} + \phi(x_\star)^T \Sigma \phi(x_\star).$$

Write a PYTHON function that implements this algorithm. Give an example of this on a synthetic set of data that includes estimation of the new target at a new input.