Machine Learning Worksheet 06

Linear Regression and Kernels

1 Sum of Squared Errors Regression

Problem 1: Let's assume we have a dataset where each datapoint, Z_n is weighted by a scalar factor which we will call T_n. We will assume that $T_n > 0$. This makes the sum of squared error function look like the following:

$$E_{\mathcal{D}}(\boldsymbol{w}) = \frac{-1}{2} \sum_{n=1}^{N} T_n \left[\boldsymbol{W}^T \phi(x_n) - Z_n \right]^2$$

Find the equation for the value of W that minimizes this error function.

Furthermore, explain what this weighting factor, T_n, does to the error function in terms of
1) the variance of the noise on the data and
2) data points for which there are exact copies in the dataset.

2 Ridge regression

Problem 2: Show that the following holds: The ridge regression estimates can be obtained by ordinary least squares regression on an augmented dataset: Augment the design matrix $\boldsymbol{\Phi}$ with p additional rows $\sqrt{\lambda}\boldsymbol{I}$ and augment $\boldsymbol{z}$ with p zeros.

3 Bayesian Linear Regression

In the lecture we made the assumption that we already knew the precision (inverse variance) for our gaussian distributions. What about when we don't know the precision and we need to put a prior on that as well as our gaussian prior that we already have on the weights of the model?

Problem 3: It turns out that the conjugate prior for the situation when we have an unknown mean and unknown precision is a normal-gamma distribution (See section 2.3.6 in Bishop). This is also true when we have a conditional gaussian distribution of the linear regression model. This means that if our likelihood is as follows:

$$p(Z \mid \mathbf{X}, W, \beta) = \prod_{n=1}^{N} \mathcal{N}(Z_n \mid W^T \phi(X_n), \beta^{-1})$$

Then the conjugate prior for both W and β is

$$p(W, \beta) = \mathcal{N}(W \mid M_0, \beta^{-1}\mathbf{S}_0) Gam(\beta \mid a_0, b_0)$$

Show that the posterior distribution takes the same form as the prior, i.e.

$$p(W, \beta \mid Z) = \mathcal{N}(W \mid M_N, \beta^{-1}\mathbf{S}_N) Gam(\beta \mid a_N, b_N)$$

.

Also be sure to give the expressions for M_N, $\mathbf{S}_N$, a_N, and b_N.

4 Facebook advertisements

You want to boost your Facebook page and therefore you book Facebook advertisements. A simple linear model for the number of new likes per week (y), depending on the money spent (x) could be:

$$\begin{aligned} y &= a_0 + a_1 x + \epsilon \\ \text{where } y &= \text{number of new likes per week} \\ x &= \text{money spent in that week, in units of 1 EUR} \\ \epsilon &= \text{normal (Gaussian) distributed fluctuations} \end{aligned}$$

After taking a lot of measurement data you fit the parameters. You find:

$$\begin{aligned} a_0 &= 10 \\ a_1 &= 5 \\ \mathbb{E}[y] &= 0 \\ \text{Var}[y] &= 4 \end{aligned}$$

The full model is therefore given by

$$\begin{aligned} y &= 10 + 5x + \mathcal{N}(0, 4) \\ &= 10 + 5x + (8\pi)^{-1/2} \exp(-x^2/8) \end{aligned}$$

Problem 4: Assume you spend no money, what is the probability that you get more than 10 likes per week?

Problem 5: Now you spend 1 EUR on advertisements. What is the expected value of likes?

5 Kernelised k-nearest neighbours

To classify the point $\vec{x}$ the k-nearest neighbours finds the k training samples $\mathcal{N} = \{\vec{x}^{(s_1)}, \vec{x}^{(s_2)}, \dots, \vec{x}^{(s_k)}\}$ that have the shortest distance $||\vec{x} - \vec{x}^{(s_i)}||_2$ to $\vec{x}$. Then the label that is mostly represented in the neighbour set $\mathcal{N}$ is assigned to $\vec{x}$.

Problem 6: Formulate the k-nearest neighbours algorithm in feature space by introducing the feature map $\vec{\phi}(\vec{x})$. Then rewrite the k-nearest neighbours algorithm so that it only depends on the scalar product in feature space $K(\vec{x}, \vec{y}) = \vec{\phi}(\vec{x})^T \vec{\phi}(\vec{y})$.

6 Radial Basis Kernels

We have a set of rules that we discussed in the lecture for proving that a kernel is a valid kernel.

Problem 7: Use these rules to show why a radial basis kernel

$$K(x_i, x_j) = exp\{-\frac{1}{2}\|x_i - x_j\|^2\}$$

is a valid kernel.

Hint: You might want to start by proving $k(x_i, x_j) = f(x_i)k_1(x_i, x_j)f(x_j)$ where k_1 is some other valid kernel.