Tutoring Session 06

Linear Classification and Kernels

1 Kernels

Problem 1: Verify that $k(X_i, X_j) = f(X_i)k_1(X_i, X_j)f(X_j)$, where k_1 is a valid kernel, is also a kernel.

Problem 2: Verify that $k(X_i, X_j) = q(k_1(X_i, X_j))$ (where q is a polynomial and k_1 is a valid kernel) and $k(X_i, X_j) = \exp(k_1(X_i, X_j))$ are valid rules for constructing a valid kernel.

Problem 3: One commonly used kernel is the Gaussian kernel, i.e.

$$k(X_i, X_j) = \exp(\frac{-\|X_i - X_j\|^2}{2\sigma^2}),$$

but remember that in this context it is not interpreted as a probability density, that's why there isn't a normalization coefficient. We can see that this is a valid kernel by expanding the square

$$\|X_i - X_j\|^2 = X_i^T X_i + X_j^T X_j + 2X_i^T X_j$$

which gives us

$$k(X_i, X_j) = \exp(\frac{-X_i^T X_i}{2\sigma^2}) \exp(\frac{X_i^T X_j}{\sigma^2}) \exp(\frac{-X_j^T X_j}{2\sigma^2})$$

Which we can derice since we know $k(X_i, X_j) = f(X_i)k_1(X_i, X_j)f(X_j)$ is a valid kernel, and we know that $k(X_i, X_j) = \exp(k_1(X_i, X_j))$ is a valid kernel.

So, by using the expanded version of the kernel from above, and expanding the middle factor as a power series, show that the Gaussian kernel equation we showed at the very top of this exercise can be expressed as the inner product of an infinite-dimensional feature vector.

Problem 4: Find an infinite-dimensional feature space $\vec{\phi}(\vec{x})$ corresponding to the Gaussian kernel, i.e. determine $\vec{\phi}(\vec{x})$ so that

$$\vec{\phi}(\vec{x})^T \vec{\phi}(\vec{y}) = \exp\left(-\frac{|\vec{x} - \vec{y}|^2}{2\sigma^2}\right).$$

(Hint: The multinomial formula turns a power of a sum into a weighted sum of products,

$$\left(\sum_{t=1}^{m} x_t\right)^n = \sum_{k_1+k_2+\cdots+k_m=n} \binom{n}{k_1, k_2, \ldots, k_m} \prod_{t=1}^{m} x_t^{k_t},$$

with $\binom{n}{k_1,k_2,\ldots,k_m} = \frac{n!}{k_1!k_2!\cdots k_m!}$.)

2 Multi-Class Classification

Problem 5: Consider a generative classification model for K classes defined by prior class probabilities $p(y = k) = \pi_k$ and general class-conditional densities $p(\phi(x)|y = k, \theta_k)$ where $\phi(x)$ is the input feature vector and $\theta = \{\theta_k\}_{k=1}^K$ are further model parameters. Suppose we are given a training set $\mathcal{D} = \{(\phi(x^{(n)}), t^{(n)})\}_{n=1}^N = \{(\phi^{(n)}, t^{(n)})\}_{n=1}^N$ where $t^{(n)}$ is a binary target vector of length K that uses the 1-of-K(hot one) coding scheme, so that it has components $t_j^{(n)} = \delta_{jk}$ if pattern n is from class $y = k$. Assuming that the data points are iid, show that the maximum-likelihood solution for the prior probabilities is given by

$$\pi_k = \frac{N_k}{N}$$

where N_k is the number of data points assigned to class $y = k$.

Problem 6: Using the same classification model as in the previous question, now suppose that the class-conditional densities are given by Gaussian distributions with a shared covariance matrix, so that

$$p(\phi(x)|y = k, \theta_k) = p(\phi(x)|\theta_k) = \mathcal{N}(\phi \mid \mu_k, \Sigma).$$

Show that the maximum likelihood solution for the mean of the Gaussian distribution for class C_k is given by

$$\mu_k = \frac{1}{N_k} \sum_{\{n|\phi^{(n)} \in C_k\}} \phi^{(n)}$$

which represents the mean of those feature vectors assigned to class C_k.

Similarly, show that the maximum likelihood solution for the shared covariance matrix is given by

$$\Sigma = \sum_{k=1}^K \frac{N_k}{N} \mathbf{S}_k$$

where

$$\mathbf{S}_k = \frac{1}{N_k} \sum_{\{n|\phi^{(n)} \in C_k\}} (\phi^{(n)} - \mu_k)(\phi^{(n)} - \mu_k)^T.$$

Thus Σ is given by a weighted average of the covariances of the data associated with each class, in which the weighting coefficients N_k/N are the prior probabilities of the classes.

Problem 7: Verify the relation $\frac{d\sigma}{da} = \sigma(1 - \sigma)$ for the derivative of the logistic sigmoid defined by $\sigma(a) = \frac{1}{1+\exp(-a)}$.

3 Hinge loss

The hinge loss is given as

$$\mathcal{L}(\mathbf{x}_i) = \max(0, 1 - y_i \tilde{z}_i),$$

where $y_i = \mathbf{x}_i^T \mathbf{w}$ is the model output and z_i the target variable ($w_0 = b$ and $x_{i,0} = 1$).

Note that in this case the computation uses class labels $\tilde{z} = 2z - 1 \in \{-1, 1\}$ instead of $z \in \{0, 1\}$.

For multiple samples $\mathbf{X}$ and respective outputs $\mathbf{y}$ and $\tilde{\mathbf{z}}$ the loss is $\mathcal{L}(\mathbf{X}) = \sum_i \mathcal{L}(\mathbf{x}_i)$.

Problem 8: Try to understand what the hinge loss does and explain it in a few of words:

4 Soft Zero-one loss

The soft zero-one loss is given as

$$\mathcal{L}(\mathbf{x}_i) = (\sigma(\beta y_i) - z_i)^2 ,$$

with $y_i = \mathbf{x}_i^T \mathbf{w}$ the model output and z_i the target variable.

Problem 9: Explain the soft zero-one loss in a few words.

Problem 10: Derive the gradient $\frac{\mathrm{d}\mathcal{L}(\mathbf{x}_i)}{\mathrm{d}\mathbf{w}}$.