Fourth week practicals in Machine learning 1 – 2024 – Paper 1

1 Multi-class Logistic Regression (September)

In the last lectures, we introduced the binary classification version of logistic regression. Here you will derive the gradients for the general case $K > 2$. Many of the preliminaries are in Bishop 4.3.4. For $K > 2$ the posterior probabilities take a generalized form of the sigmoid called softmax:

$$y_k = p(\mathcal{C}_k|\boldsymbol{\phi}) = \frac{\exp(a_k)}{\sum_i \exp(a_i)},$$

where $a_k = \mathbf{w}_k^T\boldsymbol{\phi}$ and $\boldsymbol{\phi}$ is short for $\boldsymbol{\phi}(\mathbf{x}) = (\phi_0(\mathbf{x}), \phi_1(\mathbf{x}), \dots, \phi_{M-1}(\mathbf{x}))^T$ with $\phi_0(\mathbf{x}) = 1$. Note that the posterior for class k depends on all the other classes i; keep this in mind when working out the derivatives for $\mathbf{w}_k$. The training set is a pair of matrices $\mathbf{\Phi}$ and $\mathbf{T}$. Each row of $\mathbf{T}$ uses a one-hot encoding of the class labelling for that training example, meaning that the n-th row contains a row vector $\mathbf{t}_n^T$ with all entries zero except for the k-th entry which is equal to 1 if data point n belongs to class $\mathcal{C}_k$. Answer the following questions:

(a) Write down $\frac{\partial y_k}{\partial \mathbf{w}_j}$ (you have already calculated this derivative for the first assignment). Use the indicator function I_{kj} (or kronecker delta), which you can also think of as the element at position (k, j) of the identity matrix; which can be also denoted as $\mathbb{I}[k = j]$.

(b) Write down the likelihood $p(\mathbf{T}|\mathbf{\Phi}, \mathbf{w}_1, \dots, \mathbf{w}_K)$ as a product over N and K. Use the entries of $\mathbf{T}$ as selectors of the correct class. Then write down the log-likelihood $\log p(\mathbf{T}|\mathbf{\Phi}, \mathbf{w}_1, \dots, \mathbf{w}_K)$.

(c) Derive the gradient of the log-likelihood $\nabla_{\mathbf{w}_j} \log p(\mathbf{T}|\mathbf{\Phi}, \mathbf{w}_1, \dots, \mathbf{w}_K)$.

(d) What is the objective function we minimize that is equivalent to maximizing the log-likelihood?

(e) Write a stochastic gradient algorithm for logistic regression using this objective function. Make sure to include indices for time and to define the learning rate. The gradients may differ in sign switching from maximizing to minimizing; don't overlook this.

(f) In practice, the above vanilla SGD is not effective. Point out a potential weakness of the above algorithm and/or suggest a possible improvement upon it.