CS-E4820 Machine Learning: Advanced Probabilistic Methods

Pekka Marttinen, Paul Blomstedt, Homayun Afrabandpey, Reza Ashrafi, Betül Güvenç, Tianyu Cui, Pedram Daee, Marko Järvenpää, Santosh Hiremath (Spring 2019)

Exercise problems, round 4, due on Tuesday, 26th February 2019, at 23:55

Please return your solutions in MyCourses as a single PDF file.

General instructions. Include the completed code templates in your solutions. Highlight your own code e.g. by enclosing it in a delimited block

```
# >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

my code

# <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
```

Problem 1. *"ML-II for a linear model."*

Fit the Bayesian linear parameter model to a given data `ex4_1_data.txt` using the ML-II approach. Optimize the hyperparameters α and β using grid search. Complete the template `ex4_1_template.py` with your own code. Make predictions for the test data using the fitted model and compute the mean squared error for test data. Also plot the data and the fitted model.

Problem 2. *"Optimizing linear model hyperparameters with validation set."*

Fit the Bayesian linear parameter model to a given data `ex4_1_data.txt`, as in Problem 1, but optimize the hyperparameters α and β by dividing the training data into training and validation sets, and selecting the values of α and β that minimize the mean squared error for the validation set.

Complete the template `ex4_2_template.py` with your own code. Make predictions for the test data using the fitted model and compute the mean squared error for test data. Plot the data and the fitted model.

Problem 3. *"Posterior of regression weights."*

Suppose $y_i = \mathbf{w}^T\mathbf{x}_i + \epsilon_i$, for $i = 1, \dots, n$, where $\epsilon_i \sim \mathcal{N}(0, \beta^{-1})$. Assume a prior

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \alpha^{-1}\mathbf{I}).$$

Use 'completing the square' to show that the posterior of $\mathbf{w}$ is given by $p(\mathbf{w}|\mathbf{y}, \mathbf{x}, \alpha, \beta) = \mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})$, where

$$\mathbf{S} = \left(\alpha\mathbf{I} + \beta\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^T\right)^{-1},$$

$$\mathbf{m} = \beta\mathbf{S}\sum_{i=1}^{n} y_i\mathbf{x}_i.$$

Problem 4. *"Poisson regression with Laplace approximation."*

Poisson regression can be used to model count data. A Poisson regression model can be defined as

$$y_i \mid \boldsymbol{\theta} \sim \text{Poisson}(\exp(\boldsymbol{\theta}^T \mathbf{x}_i)) \tag{4.1}$$

$$\boldsymbol{\theta} \sim \mathcal{N}(\mathbf{0}, \alpha^{-1}\mathbf{I}) \tag{4.2}$$

where y_i are the observed counts, $\mathbf{x}_i$ the related covariates, $i = 1, \ldots, N$, and $\boldsymbol{\theta}$ are the regression weights. In this exercise, we approximate the posterior $p(\boldsymbol{\theta}|\mathbf{y})$ using the Laplace approximation.

(a) Derive the gradient $\nabla \log p(\boldsymbol{\theta}|\mathbf{y})$ and the Hessian $\mathbf{H} = \nabla\nabla \log p(\boldsymbol{\theta}|\mathbf{y})$ needed for the Laplace approximation.

(b) Write the Laplace approximation as the density of a Gaussian distribution. What is the mean and the covariance matrix of this distribution?

(c) Compare the Laplace approximation to the true posterior (computed using numerical integration), in a case where we have one-dimensional covariates only. Use data given in the file `ex4_4_data.txt` and hyperparameter value $\alpha = 10^{-2}$. Plot the two posteriors and the true value $\theta = \pi/4$ used to generate the data. Also plot the data with the regression line $\hat{y}_i = \exp(\hat{\theta} x_i)$ using the MAP estimate $\hat{\boldsymbol{\theta}}$. A Python template `ex4_4_template.py` is attached to help with this.