Assessed Coursework Project (Lab 4)

CM50268 Bayesian Machine Learning 2020-2021

April 2021

1 Overview

- **Marks**: 55% of overall unit
- **Release date**: April 20 2021
- **Submission due**: 6pm, May 17 2020
- **Submission type**: project report (no more than 2,000 words) + your own Python code
- **Plagiarism**: both the code and the report will be cross checked by the plagiarism detector. Submissions with high similarity score will be manually checked, and the confirmed pairs will be reported.

The final piece of assessed coursework involves the evaluation of Bayesian modelling methods on a real multivariate regression task. The guiding objectives are to derive a good predictor for data derived from an "energy efficiency" data set, and to estimate which of the input variables are relevant for prediction.

In particular, the exercise focuses on approximating (and averaging over) posterior distributions using the Hamiltonian Monte Carlo stochastic method (an implementation of this algorithm is supplied). Your experiments will be based mainly on existing (or supplied) analytic code and techniques you have already applied. You will of course need to write all the relevant code to process the data, apply the methods appropriately, extend them in places, and ultimately calculate and output the necessary results.

A key part of the assessment is to compile, present and critique all those results effectively within a "project report" document. For this exercise, your code will not be assessed; marks will be awarded based solely on the content of the report.

2 Data

You will be analysing the "Energy efficiency" data set, originally from the University of Oxford, and now made available at the UCI Machine Learning Repository.

This multivariate data set contains 768 examples and comprises eight input variables $x_1, x_2, \cdots, x_8$ presenting some basic architectural parameters for buildings (e.g. "Roof Area" and "Glazing Area") with the intention of predicting a ninth target variable y, the required "Heating Load". This can be

considered a real-value variable, suitable for standard regression modelling (with the usual Gaussian noise model).

The data has been pre-processed and equally split into two specific data sets for your use:

- Training set: ee-train.csv
- Test set: ee-test.csv

Both csv files have a header row (labelling each variable), with the first 8 columns representing the input variables $(x_1, x_2, \cdots, x_8)$ and the final column being the target variable (y "Heating Load"). For the purposes of modelling, you may find it useful to:

- Add a "bias" input (a constant) to your models
- Standardise the other inputs to mean zero and standard deviation one

For model training, only ee-train.csv should be used, and ee-test.csv should be reserved purely for assessing model performance.

3 Summary of objectives and mark scheme

The principal objectives are focused on the predictive modelling of the energy efficiency data and take the form of a series of largely sequential sub-tasks. These are, in summary:

1. Undertake an initial exploratory analysis of the training data and summarise. [5 marks]
2. Apply the standard Bayesian linear regression model (Lecture 3), then:
   (a) Using Type-II maximum likelihood (Lecture 4) to estimate "most probable" values for hyper-parameters. [5 marks]
   (b) Using Variational Inference (Lecture 9) with simple 'Mean-Field Theory' factorisation (Lecture 10) to estimate "most probable" values for the hyper-parameters. [5 marks]
   (c) Along with task 2(b), derive the corresponding variational approximation of the joint posterior distribution for the hyper-parameters. [5 marks]
3. Familiarise yourself with the use of the Hamiltonian Monte Carlo (HMC) algorithm (Lecture 8), initially verifying the HMC implementation on a simple Gaussian example. [5 marks]
4. Apply HMC to sample weights and the hyper-parameters of the standard Bayesian regression model. [10 marks]
5. [Bonus] Modify the HMC sampling framework as a classifier to address a reformulation of the problem. [5 marks]
6. Document all results in a coherent and structured report, assessment will be based on
   - Overall quality of the report. [10 marks]
   - Clear and informative presentation of the figures and/or numerical tables. [5 marks]
   - Some degree of critical review and analysis of the project [5 marks]

4 Tasks in detail

4.1 Exploratory analysis [5 marks]

Undertake an initial exploratory analysis of the eight data variables and summarise appropriately. This need not be particularly extensive, but it should demonstrate that you have undertaken some degree of "due diligence" with respect to the data set. In particular, you should focus on identifying which of the input variables (if any) might be expected to be useful for predicting heating load, and which might be irrelevant. You may also wish to comment on the apparent linearity, or otherwise, of the problem.

As the final part of the exploratory analysis, establish a predictive "baseline" by fitting a linear model to the training set by least-squares, and assessing its prediction accuracy on both train and test sets.

This section of your report should include:

- Initial observations as to the difficulty of the task, its linearity etc.
- Your comment on the likely relevance of the variables for predicting "Heating Load"
- Appropriate graphs/charts as evidence to support the above
- Detail of the accuracy of the least-squares linear model, on both train and test sets, in terms of root-mean-square-error (RMSE)

4.2 Bayesian linear regression [15 marks]

Consider a standard linear regression model with unknown coefficient set $\mathbf{w}$. Some modelling guideline/hints:

- The problem can be modelled using an additive Gaussian noise $\mathcal{N}(0, \sigma_\epsilon)$
- $\mathbf{w}$ is assumed to have a Gaussian prior $\mathcal{N}(0, \sigma_{\mathbf{w}})$
- Easy to define an unknown hyper-parameter set $\alpha = (\sigma_\epsilon, \sigma_{\mathbf{w}})$
- If we denote the observation data as D, the posterior we want to estimate can be written as $p(\alpha, \mathbf{w}|D)$
- Please follow the methodology outlined in Lecture 4 (and maybe also Lab 2, task 1b) to solve task (a).
- Please follow the methodology outlined in Lecture 10 to solve task (b) and (c). Please note that, although looks similar, the problem described here is not the same as the one described in Example 2 of Lecture 10. You should only follow the methodology described in Lecture 10 to solve the problem.

For consistency, and the avoidance of error, it is strongly recommended that you use only natural logarithms for hyper-parameter scales. That is, you would use `numpy.exp()` and `numpy.log()` functions to convert (or to convert back) the logarithm terms.

This section (all 3 tasks (a-c)) of your report should at least include:

- A visualisation (e.g. using plt.contourf) of the posterior distribution.
- The 'most probable' values of the parameters of interest, ideally marked on the visualisation.
- RMSE of your model predictions on the test set.

4.3 Verify HMC on a simple Gaussian example [5 marks]

Code to implement HMC is supplied in the module `hmc_lab4.py` and there is a simple demonstration of its usage in the notebook `demo_hmc.ipynb` along with a one page instruction `hmc_lab4Spec.pdf`. You should try and complete this exercise by the end of the lab class.

Apply the `hmc_lab4.sample` function to generate samples from a simple correlated Gaussian in two dimensions. This is an artificial challenge, of course, as we can sample directly from such a Gaussian. However, it enables us to check our code is working and visualise its performance.

To apply HMC, you will need to write appropriate functions `energy_func` and `energy_grad` to pass to sample. The former is the negative log probability of the 2 dimensional variables under the Gaussian (easily obtained directly from `scipy.stats`). The latter is the gradient of that function, returning an array containing the partial derivatives with respect to two dimensional variables (see the notebook demo for an example). You will need to work those derivatives out and code the `energy_grad` function explicitly.

In `demo_hmc.ipynb`, a value of `L` of 25 should be fine for this simple distribution, but you will need to adjust the `epsilon0` parameter appropriately following the guideline in `hmc_lab4Spec.pdf`.

It is highly recommended that you always test the consistency of your functions by setting `checkgrad=True` when calling `sample`. This will compute an estimate of the gradient using numerical techniques, and compare it with your analytical calculation. It is a numerical approximation, so there will always be small differences, but anything large (one part in 10^6 or greater perhaps) may suggest an error in your working or your code.

This section of your report, you should:

- Design your own 2 dimensional Gaussian example
- Verify and demonstrate (with appropriate figures or numerical tables) that your HMC works as expected
- Report the values of `R`, `L` and `epsilon0` that you used to obtain your presented results
- Report your designed functions `energy_func` and `energy_grad`

4.4 Apply HMC to the Linear Regression Model [10 marks]

Apply HMC to obtain samples from the joint posterior over linear regression coefficients (weights) $\mathbf{w}$, hyperparameter set α for the linear regression model on the energy efficiency training data.

This section of your report, you should:

- Demonstrate (with appropriate figures or numerical tables) that your HMC works as expected
- Report the values of `R`, `L` and `epsilon0` that you used to obtain your presented results

- Report the optimal values of the unknown terms
- Evaluate RMSE of prediction in test set
- Provide your insight and analysis supported by figures and/or numerical tables. You could also compare the different algorithms you have used so far in this project.

4.5 Bonus: apply HMC as a Classifier [5 marks]

The objective is to predict a "high" heating load. That is, (artificially) transform the target data to a binary class indicator by labelling all cases with original target variable ("Heating Load") greater than 23.0 as "positive". That particular threshold is chosen simply because it approximately splits the training set in half.

You will need to modify `energy_func` and `energy_grad` to use the Bernoulli likelihood with a sigmoid link function (essentially, this is now a logistic regression model). The prior should remain the same. You should be able to obtain a test mis-classification rate of as little as 1%.

This is an bonus task, you could demonstrate the results and write analysis supported by figures and/or numerical tables.

4.6 Your overall report quality [20 marks]

The mark scheme of your overall report quality has been listed in the last item of section 3. In your report summary, please write a generous paragraph (or two) summarising what you have learned in respect of both:

- The specific data set under study, the suitability of linear models for prediction and, in particular, the relevance of the different variables
- Your observations on the effectiveness of the various methods applied in this project.

You should submit both your report and the code (ideally jupyter notebook) via Moodle by the specified due date. The report should be self-contained with sufficient information to allow any reader to replicate the experiments, though be concise and remember the 2,000 word limit. (Note that graphs and listings of code do not contribute to the limit. Use of multiple explanatory graphs is indeed encouraged!)

- Please follow the instructed order of objectives in Section 3 to write your report
- You should make use of additional text to make clear what is being presented. Do not assume that the reader has the task specification (i.e. this document) available to cross-reference.
- Please be as clear as possible in your presentation! With the best will in the world, marks cannot be awarded for content that cannot be understood :).