A4: Bayesian Logistic Regression Report

ROB313 Intro to Learning from Data
Instructor: Trefor Evans
Prepared by:

Li, En Xu
1004028759

April 11, 2020

1 Objectives

The overall objective of this assignment is to learn about Bayesian Inference as well as some machine learning topics such as fairness in classifiers. A Bayesian inference model to classification dataset, iris, is built with laplace approximation. Different prior variances are used to experiment with the model. In addition, importance sampling approach is taken to build the Bayesian classifier with Gaussian proposal distribution. Validation dataset is used to select the best variance and sample size for the sampler. Lastly, a literature review is done on the fairness in machine learning models. In summary, this assignment provides practical perspectives on how Bayesian models work as well as importance sampling.

2 Code Structure and Strategies

The code used to do this assignment is very modular and is structured as the following:

Q1a: Bayesian Inference using Laplace Approximation Load the iris dataset and train the model 3 times using one of the variance [0.5,1,2]. For each training, a while loop is used to update weights until a convergence threshold given by user is met.

Q1b: Importance Sampling call **_importance_sampling_train_val** to try different sets of proposal variance and sample size and choose the optimal hyperparameter. Then **_importance_sampling_test** is called to compute the test loss and accuracy while visualizing the posterior.

Strategies The strategies include making the functions modular. For example, the general **_log_likelihood** function can be used in many other functions for training and evaluating. In this way, there is no need to re-write it again, and we can call the previously written function instead. Furthermore, we can take advantage of numpy's vectorization to save time in computing rather than using iterative methods. This way will make the code run much faster.

3 Bayesian Inference (Q1a)

Three sets of prior variances, [0.5, 1.0, 2.0] are experimented with the Iris dataset. The log marginal likelihoods of the three models are reported in Table 1. Note that the marginal likelihood is approximated with the Laplace approximation using the following equation.

$$log(Pr(y|X)) \approx log(Pr(y|X,w)) + log(Pr(w)) + \frac{M}{2}log(2\pi) - \frac{1}{2}log(det(-H))$$

Table 1: Log Marginal Likelihood for Variance {0.5,1,2}

Prior Variance	**Log Marginal Likelihood**
0.5	-74.82358184
1.0	-74.50607931
2.0	-74.81392016

The most complex model usually has the smallest marginal likelihood which means a most negative log likelihood. The small marginal likelihood indicates that it spreads its probability mass very widely (predict that everything is possible). Therefore, from the table above, I can conclude that the model with prior variance 0.5 is the most complex model.

4 Importance Sampling (Q1b)

The predictive posterior likelihood is computed with importace sampling method using the following equation.

$$Pr(y^*|\mathbf{y},\mathbf{X},\mathbf{x}^*) \approx \sum_{i=1}^{S} Pr(y^*|\mathbf{w^{(i)}},\mathbf{x}^*)[\frac{r(\mathbf{w}^{(i)})}{\sum_{j=1}^{S} r(\mathbf{w}^{(j)})}]$$

Note that $r(\mathbf{w}) = \frac{Pr(\mathbf{y}|\mathbf{w},\mathbf{X})*Pr(\mathbf{w})}{q(\mathbf{w})}$ and $q(\mathbf{w})$ is the proposal distribution. A Gaussian distribution centred at the MAP solutions computed from the previous part will be the proposal distribution used. The variance and the sample size will be 2 hyperparameters to experiment with training and validation dataset. In short, variance of [0.5, 1.0, 2.0, 5.0] and sample size of [10, 50, 100, 500, 1000] are evaluated on the validation set. The validation loss and accuracy are shown below.

Table 2: Validation Loss

	size = 10	**size = 50**	**size = 100**	**size = 500**	**size = 1000**
var = 0.5	0.4826	0.4806	0.5119	0.4916	0.4935
var = 1.0	0.5387	0.5177	0.5065	0.4887	0.4942
var = 2.0	0.5584	0.5216	0.4984	0.4948	0.5097
var = 5.0	0.6945	0.6658	0.5703	0.5694	0.5106

Table 3: Validation Accuracy

	size = 10	**size = 50**	**size = 100**	**size = 500**	**size = 1000**
var = 0.5	70.97%	70.97%	67.74%	67.74%	67.74%
var = 1.0	67.74%	70.97%	67.74%	67.74%	67.74%
var = 2.0	67.74%	67.74%	74.19%	67.74%	70.97%
var = 5.0	64.51%	67.74%	77.42%	67.74%	70.97%

As seen, with variance=0.5 and sample size=50, the validation loss (negative log likelihood) is achieved. Therefore, I will choose the proposal distribution to be the multivariate Gaussian centred at the MAP solution with variance of 0.5. In addition, a sample size of 50 will be used to approximate. The complete result on the test set is shown in the following Table.

Table 4: Summarized Final Result

Proposal Variance	**Sample Size**	**Validation Loss**	**Validation Accuracy**	**Test Loss**	**Test Accuracy**
0.5	50	0.4806	70.97%	0.4740	73.33%

As seen, the normalized validation and test loss and accuracy are very close to each other. However, since it's a random sampling procedure, a small sample size may not always yield a good result because sometimes it can get unlucky. The result can also be visualized by plotting the posterior. The following figures show an overlapping of the posterior and the chosen proposal distribution on some of the weights.

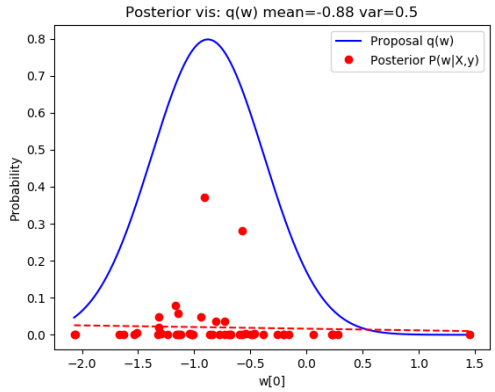

Figure 1: Posterior Visualization 1

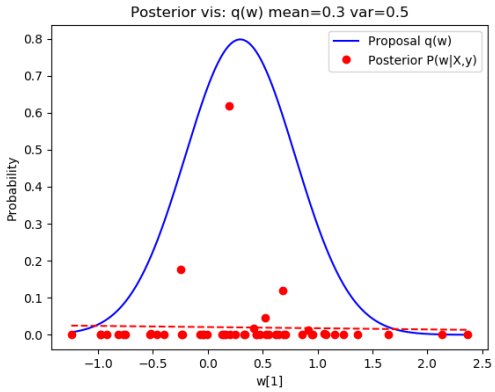

Figure 2: Posterior Visualization 2

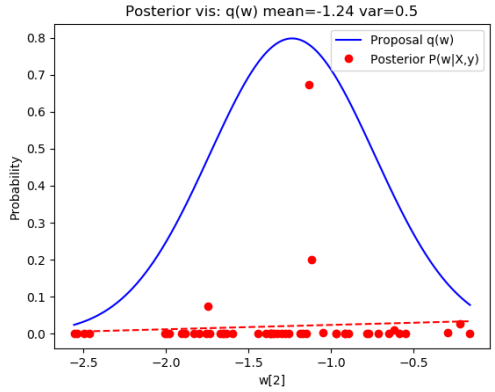

Figure 3: Posterior Visualization 3

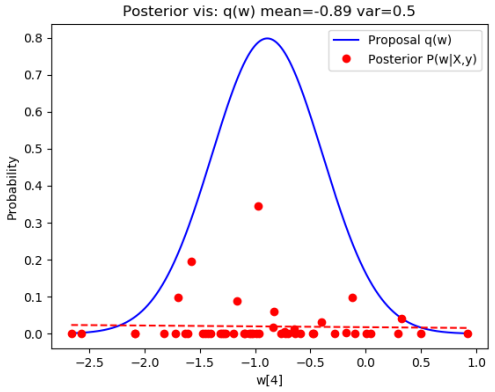

Figure 4: Posterior Visualization 4

As seen from the visualization, the proposal distribution does show a trend of the posterior. Areas of dense probability correspond to the peak of the chosen proposal distribution.

5 Machine Learning Literature Review

The following literature view is done on *Fairness Constraints: Mechanisms for Fair Classification*[1]. With the growing use of machine learning solutions to the real world problems, fairness of such solutions is a serious topic to consider. This article aims to promote fairness in machine learning models by reducing disparate treatment and disparate impact. In other words, the paper tries to propose methods to ensure the model's decision will not be based on sensitive attributes and will not hurt people with certain sensitive attributes, such as, race and sex.

The paper has provides two ways for training fair models:

1. maximizes accuracy subject to fairness constraints and enable compliance with disparate impact doctrine in its basic form (i.e., the p%-rule)
   minimize $L(\theta)$
   constraint $\frac{1}{N}\sum_{i=1}^{N}(\mathbf{z}_i - \bar{\mathbf{z}})d_\theta(\mathbf{x}_i) \leq \mathbf{c}$

2. maximizes fairness subject to accuracy constraints and ensure fulfilling the business necessity clause of disparate impact
   minimize $|\sum_{i=1}^{N}(\mathbf{z}_i - \bar{\mathbf{z}})d_\theta(\mathbf{x}_i)|$
   constraint $\mathbf{L}(\theta) \leq (1+\gamma)\mathbf{L}(\theta^*)$

Experiments of both approaches were firstly tested on synthetic data. They were able to extract sensitive attributes thus successfully eliminate disparate treatment. While testing on the real-world data, the classifiers with accuracy constraint remove non-protected users from the positive label and add protected users to the positive label. On the other hand, classifiers with accuracy constraint are able to label entries without sensitive attributes to the negative class and the rest to the positive class. Finally, the paper mentioned one of the area for improvement is to show more rigorous proof on the correlation between covariance and p%-rule.

The paper explained the two approaches thoroughly to the readers; however, I believe there should be more metrics on the fairness of the model besides the p%-rule. And one of the biggest weakness of this approach in my opinion is the assumption that the every training data contains some sensitive attributes. The method assumes that the training data is biased in the first place. I think it will be more appropriate to look for bias in the data while keeping in mind that there may or may not exist bias.

6 Appendix: Code

The documented code can be accessed via GitHub public repo: https://github.com/thomas-enxuli/SGD-Logistic-Regression

References

[1] M. Zafar, I. Valera, M. Rodriguez, and K. P. Gummadi, "Fairness constraints: A mechanism for fair classification," 07 2015.