Automated Prior Elicitation from Large Language Models for Bayesian Logistic Regression

Henry Gouk[1] **Boyan Gao**[2]

[1]School of Informatics, University of Edinburgh
[2]University of Oxford

Abstract We investigate how one can automatically retrieve prior knowledge and use it to improve the sample efficiency of training linear models. This is addressed using the Bayesian formulation of logistic regression, which relies on the specification of a prior distribution that accurately captures the belief the data analyst, or an associated domain expert, has about the values of the model parameters before having seen any data. We develop a broadly applicable strategy for crafting informative priors through the use of Large Language Models (LLMs). The method relies on generating synthetic data using the LLM, and then modelling the distribution over labels that the LLM associates with the generated data. In contrast to existing methods, the proposed approach does not require a substantial time investment from a domain expert and has the potential to leverage access to a much broader range of information. Moreover, our method is straightforward to implement, requiring only the ability to make black-box queries of a pre-trained LLM. The experimental evaluation demonstrates that the proposed approach can have a substantial benefit in some situations, at times achieving an absolute improvement of more than 10% accuracy in the severely data-scarce regime. We show that such gains can be had even when only a small volume of information is elicited from the LLM.

1 Introduction

Logistic regression is a ubiquitous method for building linear classifiers in machine learning, and a useful tool for data analysis in a variety scientific disciplines. Although the most common approach for model fitting relies on finding a point estimate of the parameters, one can also take a Bayesian approach and compute a posterior distribution over model parameters. While much more computationally expensive, the Bayesian approach is known for providing good uncertainty estimates—which can be useful for qualifying data analysis conclusions and incorporating additional sources of information. Moreover, it also has the potential to improve the sample efficiency, if an informative prior distribution is used. For example, in the transfer learning and meta-learning settings, it has been shown that fitting a prior on related auxiliary tasks can lead to substantial improvements in sample efficiency (Rothfuss et al., 2021; Zhang et al., 2021; Riou et al., 2023; Shwartz-Ziv et al., 2022)

However, there is little effort in the machine learning community devoted to constructing informative priors for logistic regression—or many other families of Bayesian models. With the notable exception of Bayesian transfer learning and meta-learning methods, the problem of crafting informative priors has mainly been addressed in the statistics community, where the goal is to elicit a prior from one, or many, domain experts (Falconer et al., 2022; Mikkola et al., 2023). Rather than making the prior informative, the goal is usually to construct a prior that is an accurate representation of an experts subjective belief about the parameters. Such priors are developed after undertaking elicitation sessions with the expert, which involves providing them with sufficient background in statistical concepts to facilitate communication of different data or parameter statistics. Overall, this is a time-consuming and imprecise process, as there are many human factors

at play. A popular line of research in the prior elicitation literature is to elicit information in *observation space*. This means that, rather that attempting to get an expert to make statements about their belief distribution for the parameters, one instead attempts to gain information about the expert's belief for various marginal and conditional distributions of the data. From this, the analyst can try to infer a distribution over the parameters that is consistent with the expert's belief for how the data is distributed. Methods for eliciting priors from human experts tend to focus on obtaining fractiles of the data distribution, and in practice this most often means quartiles (see, e.g., the discussion in Bockting et al. (2024)). In the case of regression models, the set of feature vectors is also often made available to the expert, and the goal of the analyst is to infer the expert's belief for the target distribution of each point (Hosack et al., 2017).

The focus of this paper is to determine the extent to which LLMs can be used in place of human experts for the purposes of prior elicitation. By using LLMs instead of human experts, we can circumvent the need for elicitation sessions, thus saving valuable time for the experts and the analyst. Moreover, in some situations the analyst will not have access to someone with the relevant expertise. While the machine learning community often makes use of Bayesian methods with uninformative priors, with the goal of producing principled estimates of uncertainty, our goal is different. We instead try to automatically construct informative priors that enable improved sample efficiency. We also follow the line of work that elicits information in observation space but, rather than trying to model the expert's conditional distribution of the labels given the training features, we also use the expert to generate novel feature vectors. We then model the likelihood of this synthetic expert-provided dataset, given the logistic regression parameters. In a conventional prior elicitation session, such a process would be impractical, but sampling features and label distributions from LLMs is relatively straightforward. The experimental evaluation shows that this approach can lead to significant improvements in sample efficiency, even when a relatively small volume of knowledge is retrieved from the LLM.

2 Prior Elicitation

The posterior over model parameters, taking into account the available data, D, and expert prior knowledge, K, is given by

$$p(\theta|D,K,\phi) = \frac{p(D|\theta,K,\phi)p(\theta|K,\phi)}{\int_\Theta p(D|\theta,K,\phi)p(\theta|K,\phi)\cdot d\theta}, \tag{1}$$

where ϕ are hyperparameters selected by the analyst. The goal of prior elicitation is to construct a prior that accurately represents the domain expert's belief about the parameters, θ. We accomplish this by modelling the prior knowledge elicited from the expert,

$$p(\theta|K,\phi) = \frac{p(K|\theta,\phi)p(\theta|\phi)}{\int_\Theta p(K|\theta,\phi)p(\theta|\phi)\cdot d\theta}. \tag{2}$$

It can be convenient to express the overall posterior as

$$p(\theta|D,K,\phi) \propto \underbrace{p(D|\theta,K,\phi)}_{\text{Data Likelihood}} \quad \underbrace{p(K|\theta,\phi)}_{\text{Knowledge Likelihood}} \quad \underbrace{p(\theta|\phi)}_{\text{Analyst Prior}}, \tag{3}$$

where $p(\theta|\phi)$ is the analysts prior belief about θ, which will usually be minimally informative. Expressing the posterior in this way makes it clear that we are modelling the knowledge elicited from the expert with the same conceptual framework that we would model any other data; $p(K|\theta,\phi)$ is another likelihood term, but it is associated with the expert knowledge, K.

```
You are an expert in the field of {field}.
Your top priority is to provide statisticians with the domain knowledge required to
analyse their data. {data description}

The dataset has the following features:
{feature name 1}: {feature description 1}
...
{feature name m}: {feature description m}

The dataset has the following target:
{target name}: {target description}
The target can take these values: {target values}.
```

Figure 1: The system prompt for generating synthetic data to be used for fitting the informative prior.

3 Bayesian Logistic Regression with LLM Priors

We follow a standard approach for specifying the data likelihood and the analyst prior for the coefficient matrix, β, and intercept vector, α. For a set $D = \{(\mathbf{x}_i, \mathrm{y}_i)\}_{i=1}^n$, where each $\mathbf{x}_i$ is a random vector of reals and each y_i is a corresponding class in $\{1, ..., C\}$, the data likelihood is given by

$$p(D|\beta, \alpha) = \prod_{i=1}^{n} \prod_{j=1}^{C} (\sigma_j(\beta \mathbf{x}_i + \alpha))^{\mathbf{1}(\mathrm{y}_i = j)}, \tag{4}$$

where $\sigma_j(\cdot)$ is the j-th component of the output of the softmax function, σ, and $\mathbf{1}(\cdot)$ is the indicator function. The analyst prior for both β and α is chosen to be a zero-mean Gaussian with identity covariance.

3.1 Eliciting Knowledge from LLMs

The knowledge we elicit from the LLM takes the form of synthetic training data. We divide the data generation process into two phases: feature synthesis, and target sampling. In both cases we take advantage of non-deterministic methods of performing inference with LLMs: sampling tokens according to their probabilities, rather than greedily selecting the token with the highest probability. In all cases, the only content shared between subsequent queries are the system and instruction prompts.

In the first phase, the LLM is prompted to generate a batch of feature vectors, and this query is repeated until the desired number of synthetic feature vectors is obtained. In the second phase, the LLM is presented with each synthetic feature vector one at a time and asked to provide a target label. This is repeated multiple times for each feature vector in order to obtain a distribution over labels. The result of this process is a set of random variables, $K = \{(\mathbf{x}_i^k, \mathbf{y}_i^k)\}_{i=1}^m$, containing synthetic feature vectors and corresponding discrete distributions representing the LLMs belief for the label.

To ensure that the LLM has sufficient information to successfully complete these tasks, we construct a system prompt, given in Figure 1 that is shared between both phases. This system prompt is constructed by filling in a template with dataset-specific meta-data. The prompt starts by instructing the LLM to be an expert in a relevant field of study, then provides a short explanation of the problem to be solved with the dataset, and finally gives a list of attributes (features and the target) with natural language descriptions. We use a guided generation technique to ensure that the continuations generated by the LLM match a JSON schema for the dataset features in the first phase, and that only valid class labels are produced in the second phase.

3.2 The Knowledge Likelihood

Prior elicitation literature has found that modelling experts' belief, not their best guess, is crucial—a paradigm sometimes referred to as supra-Bayesian methods (Mikkola et al., 2023).

We assume independence between the different synthetic data points, and that the label distributions follow a Dirichlet distribution parameterised by the logistic regression classifier,

$$p(K|\beta, \alpha, \gamma, \delta) = \prod_{i=1}^{m} \mathrm{Dir}\left(\mathbf{y}_i^k \,\middle|\, \gamma\mathbf{1} + \delta\sigma(\beta\mathbf{x}_i^k + \alpha)\right). \tag{5}$$

Across our experiments we found that a reliable choice for γ is 0.5, and good values for δ were typically in the range $[0.5, 2]$. These hyperparameters allow the analyst to specify the belief about the expert's beliefs.

3.3 Implementation Details

We implement our Bayesian Logistic Regression model in python using the PyMC library (Abril-Pla et al., 2023), performing inference with the NUTS Markov Chain Monte Carlo method (Hoffman et al., 2014), as this allows us to obtain arbitrarily close approximations to the true posterior and posterior predictive distributions. All experiments use eight billion parameter instruction-tuned variant of Llama-3[1], and the vLLM server (Kwon et al., 2023) is used for performing inference. We separately standardise the real and synthetic data, with the standardisation statistics computed on the real data being used to transform the test examples. Performing the standardisation separately adds robustness to the LLM providing features that are linearly transformed compared to the real features. This can happen, for example, when different units are used, or when time is given as an offset from some fixed point. Code for reproducing the experiments is available in our GitHub repository.[2]

4 Experiments

Experiments are conducted to demonstrate the relationship between the volume of real and synthetic data, and the performance of the models on held-out real data. See Appendix Appendix A for information on the datasets.

4.1 Sample Efficiency

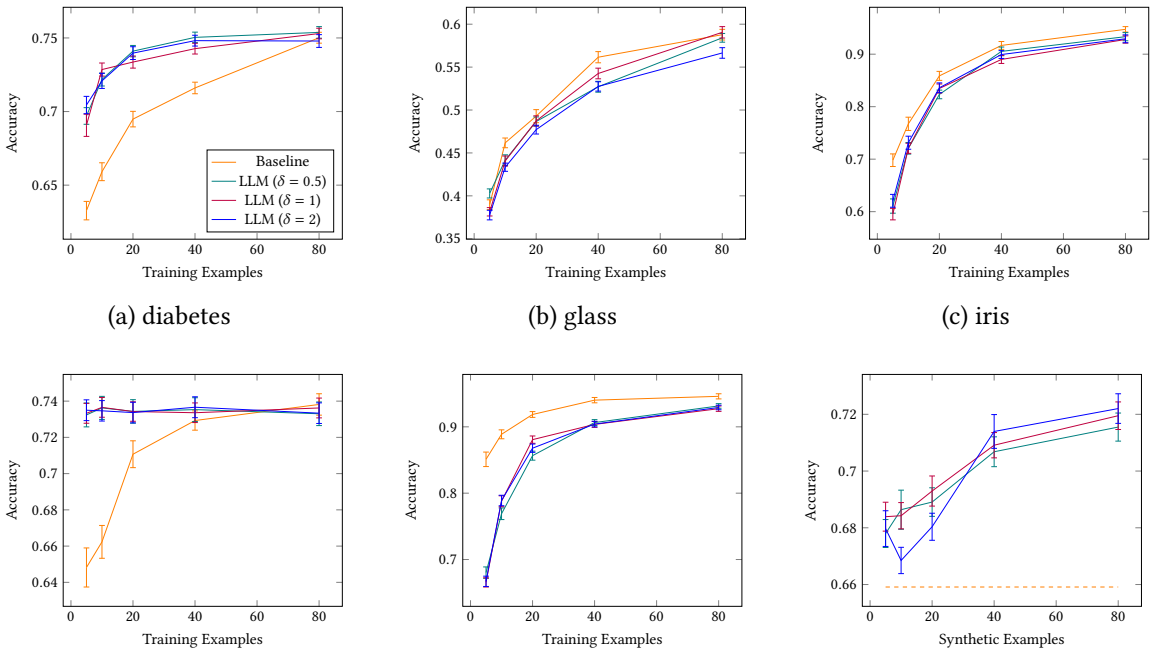
The first set of experiments investigates the sample efficiency of using a conventional Bayesian Logistic Regression model with an uninformative prior, compared to our LLM-based prior elicitation framework. For each dataset, we train the two methods with varying amounts of real training data ranging from five examples to 80 examples. For the informative prior, we use a fixed set of 80 examples generated by the LLM, and we explore several different values for δ. Accuracies are estimated using ten repetitions of 5-fold cross-validation. The results are given in Figure 2. From these plots, we can see that on the diabetes and survival datasets, the LLM is able to provide very informative priors, leading to substantial improvements in performance in the severely limited data setting. We note that setting $\delta = 0$ in our method recovers the baseline as a special case, so when using it in practice it will fail more gracefully than these plots indicate.

4.2 Impact of the Volume of Elicited Data

The second set of experiments investigates the impact of the volume of elicited synthetic data on the performance of the model when evaluated on real data. For each dataset, we train the model

[1] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

[2] https://github.com/henrygouk/llm-prior



(a) diabetes (b) glass (c) iris

(d) survival (e) vote (f) diabetes (prior samples)

Figure 2: (a–e) Plots showing the sample efficiency on each dataset, where accuracies are estimated using 10 repetition of 5-fold cross-validation, with error bars indicating the standard error; (f) A plot showing the impact of different volumes of synthetic data on the performance of models with different choices for δ.

with a fixed amount of real training data, and varying amounts of synthetic training data. We explore several different values for δ. Accuracies are once again estimated using five repetitions of 5-fold cross-validation. The results are given in Figure 2f. From these results, we can see that even with a small amount of synthetic data one can gain a measurable improvement in performance over the baseline that does not use the synthetic data. This provides some evidence that we are not just replacing a time consuming prior elicitation session with a compute intensive LLM sampling procedure; the total cost will be only a few thousand tokens.

5 Conclusion

In this paper, we explore an automated approach for crafting informative priors for Bayesian logistic regression using Large Language Models (LLMs). This method contrasts with traditional expert-driven prior elicitation, offering a broader range of information without the need for extensive expert involvement. Our experiments demonstrate significant improvements in accuracy in some cases, especially in data-scarce scenarios, highlighting the potential of LLMs to replace human experts for prior elicitation in resource-constrained setting, thus saving time and leveraging extensive pre-existing knowledge.

5.1 Limitations and Broader Impact

The main technical limitation of our work is that we have not explored model selection strategies to tune the hyperparameters, δ and γ. We would not currently recommend deploying this method in practice, as the reliance on LLMs could lead to unintended consequences, due to the broader lack of understanding on their limitations in various contexts.

Acknowledgements. This project was supported by the Royal Academy of Engineering under the Research Fellowship programme.

References

Abril-Pla, O., Andreani, V., Carroll, C., Dong, L., Fonnesbeck, C. J., Kochurov, M., Kumar, R., Lao, J., Luhmann, C. C., Martin, O. A., et al. (2023). Pymc: a modern, and comprehensive probabilistic programming framework in python. *PeerJ Computer Science*, 9:e1516.

Bockting, F., Radev, S. T., and Bürkner, P.-C. (2024). Simulation-Based Prior Knowledge Elicitation for Parametric Bayesian Models. *arXiv:2308.11672*.

Falconer, J. R., Frank, E., Polaschek, D. L. L., and Joshi, C. (2022). Methods for Eliciting Informative Prior Distributions: A Critical Review. *Decision Analysis*, 19(3):189–204.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188.

German, B. and Spiehler, V. (1987). Glass identification database. *UC Irvine Machine Learning Repository*.

Haberman, S. J. (1976). Generalized residuals for log-linear models. In *Proceedings of the 9th international biometrics conference*, pages 104–122.

Hoffman, M. D., Gelman, A., et al. (2014). The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623.

Hosack, G. R., Hayes, K. R., and Barry, S. C. (2017). Prior elicitation for Bayesian generalised linear models with application to risk control option assessment. *Reliability Engineering & System Safety*, 167:351–361.

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J. E., Zhang, H., and Stoica, I. (2023). Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Mikkola, P., Martin, O. A., Chandramouli, S., Hartmann, M., Pla, O. A., Thomas, O., Pesonen, H., Corander, J., Vehtari, A., Kaski, S., Bürkner, P.-C., and Klami, A. (2023). Prior knowledge elicitation: The past, present, and future. *arXiv:2112.01380*.

Riou, C., Alquier, P., and Chérief-Abdellatif, B.-E. (2023). Bayes meets Bernstein at the Meta Level: An Analysis of Fast Rates in Meta-Learning with PAC-Bayes. *arXiv:2302.11709*.

Rothfuss, J., Fortuin, V., Josifoski, M., and Krause, A. (2021). PACOH: Bayes-Optimal Meta-Learning with PAC-Guarantees. In *International Conference on Machine Learning*.

Schlimmer, J. C. (1987). *Concept acquisition through representational adjustment*. University of California, Irvine.

Shwartz-Ziv, R., Goldblum, M., Souri, H., Kapoor, S., Zhu, C., LeCun, Y., and Wilson, A. G. (2022). Pre-Train Your Loss: Easy Bayesian Transfer Learning with Informative Priors. In *Advances in Neural Information Processing Systems*.

Smith, J. W., Everhart, J. E., Dickson, W., Knowler, W. C., and Johannes, R. S. (1988). Using the adap learning algorithm to forecast the onset of diabetes mellitus. In *Proceedings of the annual symposium on computer application in medical care*, page 261. American Medical Informatics Association.

Zhang, X., Meng, D., Gouk, H., and Hospedales, T. M. (2021). Shallow bayesian meta learning for real-world few-shot recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Submission Checklist

1. For all authors...

   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] The claims are verified in the experiments section.

   (b) Did you describe the limitations of your work? [Yes] See the final section of the main paper.

   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See the final section of the main paper.

   (d) Did you read the ethics review guidelines and ensure that your paper conforms to them? `https://2022.automl.cc/ethics-accessibility/` [Yes] They have been read.

2. If you ran experiments...

   (a) Did you use the same evaluation protocol for all methods being compared (e.g., same benchmarks, data (sub)sets, available resources)? [Yes] See the supplied code. All methods had the same resources.

   (b) Did you specify all the necessary details of your evaluation (e.g., data splits, pre-processing, search spaces, hyperparameter tuning)? [Yes] There are scripts to completely reproduce our results in the code provided.

   (c) Did you repeat your experiments (e.g., across multiple random seeds or splits) to account for the impact of randomness in your methods or data? [Yes] We used repetitions and cross validation.

   (d) Did you report the uncertainty of your results (e.g., the variance across random seeds or splits)? [Yes] We use standard error of the mean.

   (e) Did you report the statistical significance of your results? [No] We report uncertainty, rather than testing on a threshold.

   (f) Did you use tabular or surrogate benchmarks for in-depth evaluations? [Yes] See the appendix.

   (g) Did you compare performance over time and describe how you selected the maximum duration? [N/A]

   (h) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] We did not use significant resources. There was less than one hour of GPU time, in total.

   (i) Did you run ablation studies to assess the impact of different components of your approach? [No] We plan to do this in an extended conference version.

3. With respect to the code used to obtain your results...

   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results, including all requirements (e.g., `requirements.txt` with explicit versions), random seeds, an instructive `README` with installation, and execution commands (either in the supplemental material or as a URL)? [Yes]

   (b) Did you include a minimal example to replicate results on a small subset of the experiments or on toy data? [Yes]

   (c) Did you ensure sufficient code quality and documentation so that someone else can execute and understand your code? [Yes]

(d) Did you include the raw results of running your experiments with the given code, data, and instructions? [No]

(e) Did you include the code, additional data, and instructions needed to generate the figures and tables in your paper based on the raw results? [Yes]

4. If you used existing assets (e.g., code, data, models)…

(a) Did you cite the creators of used assets? [No] The links to datasets are given in the GitHub repo where the datasets are stored, rather than the paper.

(b) Did you discuss whether and how consent was obtained from people whose data you're using/curating if the license requires it? [N/A]

(c) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you created/released new assets (e.g., code, data, models)…

(a) Did you mention the license of the new assets (e.g., as part of your code submission)? [No] We will consider this.

(b) Did you include the new assets either in the supplemental material or as a URL (to, e.g., GitHub or Hugging Face)? [Yes]

6. If you used crowdsourcing or conducted research with human subjects…

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

7. If you included theoretical results…

(a) Did you state the full set of assumptions of all theoretical results? [N/A]

(b) Did you include complete proofs of all theoretical results? [N/A]

A Dataset Information

We use the five datasets summarised in Table Table 1. All the datasets contain features that can be interpreted numerically. E.g., as real valued, count data, or as binary or trinary (binary and missing) indicators.

Table 1: Descriptions of the datasets used throughout our experiments.

Dataset	Domain	Features	Classes
diabetes (Smith et al., 1988)	Medicine	8	2
glass (German and Spiehler, 1987)	Forensics	9	6
iris (Fisher, 1936)	Botany	4	3
survival (Haberman, 1976)	Medicine	3	2
vote (Schlimmer, 1987)	Politics	16	2