Final Report Review: Multi-task Gaussian Processes with Task-Dependent Kernels

Summary of Report:

This project is to some extent in two parts. First, multi-task (multi-output) Gaussian processes with task-dependent RBF kernels are implemented in GPyTorch, and their improvement in certain situations over the task-homogeneous Kronecker method of multi-task GP models is explored via experiment. Then, pivoting from implementation to application, the use of auxiliary but correlated information to an objective in Bayesian optimization is explored through this task-dependent multi-output GP model. Future work is described as including Matérn kernels in the implementation, both for Matérn- Matérn cross covariance and Matérn-RBF, as well as the implementation of structured kernel interpolation for efficiency (the existing Kronecker approach has built-in inference efficiency based on its limited structure)

Review:

This report was well-structured and clear. It had a solid background section assuming some general knowledge of GPs and strong motivation for its methods and approaches. Methodology was generally detailed and left the reader with what felt like a full understanding of what was undertaken and what was achieved. Aside from a few odd design choices in figures, experiments told a clear story and each seemed to answer some natural question following the last.

The authors did not achieve every goal set forth in their proposal and midterm report, namely the implementation of cross-covariance functions aside from RBF-RBF and application to real-world data, but explained why in their coversheet and substituted those goals with an application to Bayesian optimization with auxiliary information. This pivot led down an interesting path, and I agree with their conclusion that this is a promising avenue of investigation.

As mentioned, this report was very clear and easy to read. It made good observations and produced interesting results, especially those reported in figure 2 regarding training data distribution in parameter space and fit quality of the multi-task implementation. The main weakness I see is perhaps the complete lack of examples that feel non-contrived. While I understand that application to complex real data such as the NAV and market price time series described in the midterm report is a challenging task, any dataset drawn from a source other than a GP model would speak volumes towards the robustness and significance of this method. Even in the context of Bayesian optimization no real problems were used as demonstration.

I would rate this report as one of high quality, good clarity, moderate originality (although the idea to explore Bayesian optimization with this method is a very good one), and moderate significance. I would consider it more significant if a real or even less contrived dataset was used as a proof of concept at either phase of investigation. However, due to the late pivot towards Bayesian optimization I still consider this a minor criticism.

Suggestions:

My first suggestion has already been described: Find some simple enough dataset or produce one from some sufficiently far removed model such that it can reasonably be considered plausible as real data, and use it in an experiment. Surely there exist real problems well-suited to Bayesian optimization with auxiliary information that could be dreamed up, even if they would have to be quickly simulated through some other means. Almost anything but using a GP to fit data from a GP would work here.

My second suggestion is about background. One thing I really appreciated from your presentation was the information about generalized cross-covariance functions, even when they are not defined in closed form. I would have really liked to see at least a mention of how they could be computed, perhaps with details included in your appendix slides as an appendix section for the report.

A small third suggestion would be to re-evaluate the presentation of your error statistics in figures 1 and 2. I personally found them difficult to parse even after understanding what they represented. Perhaps a format where color does more work towards immediate understanding would help.

Minor edits and tweaks:

Typo: in experiment 2, you describe the models being fit “as in experiment 2”. I assume you mean as in experiment 1.

Background: in your description of the Expected Improvement acquisition function (eq. 12), phi and Phi are not defined. While I’m aware from class that they’re the CDF and PDF of the standard Gaussian, that detail should probably be referenced somewhere (maybe even further detail on the EI function and Bayesian optimization in an appendix?).

In figure 4 it seems visually obvious that your GP prior had mean zero but your objective function did not. Did that affect the optimization? Might it have been more informative to align those two quantities?

Technical Questions:

In the Bayesian optimization problem, you model T functions with a multi-task GP. Only one function is the objective, but all presumably contain some information about its value. Did you consider adapting the acquisition function to this situation in any way? Or is the idea that the information gained from auxiliary function values helping at the level of the GP model? As a follow-up question, how does this fit in with existing work on multi-task Bayesian optimization (say, Swersky et al., NeurIPS 2013)?

Would you expect your Bayesian optimization scheme to improve further as the number of auxiliary inputs increases in the presence of noise? What about in the case where non-RBF kernels were used and the cross-covariance functions became less closely tied together in functional form?