Heilmeier's Criteria

- What are you trying to do? What is the problem? Why is it hard?

  Implement multi-kernel learning on top of GPyTorch. There's currently no unified (i.e. within a larger library) implementation of multi-kernel learning for Gaussian processes. Multi-kernel learning is a new problem with limited approaches and adjustments will need to be made to GPyTorch's base code for reliable and flexible implementation.

- How is it done today, and what are the limits of the current practice?

  It isn't done today. Melkumyan has a method, but there is not a publically available implementation.

- What's new in your approach and why do you think it will be successful?

  GPyTorch's implementation of multi-task GPs allows for only one kernel function that gets applied to each task - we want to extend this flexibility to allow for RBF kernels with differing length scales for each of the output tasks.

  We believe that this will be successful given prior research indicating the strength of the method and knowledge of real-world data that behaves in a fashion that reflects GPs with different kernels for each of the outputs (i.e. PIMCO CEF).

- Who Cares?

  There's obivous applications to this (finance, geology, meteorology, etc) but it also builds a foundation for further theory - what about $\mathbb{R}^2 \to \mathbb{R}^2$ with distinct kernels for each task? What about increased computational efficiency for the exact and approximate inference with the resulting kernel structure?

- If you're successful what difference will it make? What impact will success have? How will it be measured?

  If successful we will have provided a tool that removes barries to entry in complex modeling using GPs. Success will open new inroads into further research using multi-task GPs.

- What are the risks and the payoffs?

  The risks for this are that the scope is limited in terms of what kernels can be selected for each task, and that the benefits of using multiple kernels does not outweigh the increase in complexity of the model.

- How much will it cost?

  Nothing

- How long will it take?

  Hopefully a month.

- What are the midterm and final criteria for success? How will progress be measured?

  Midterm: Working implementation of multi-task learning with RBF kernels. Validated on synthetic data. If implementation goes smoothly, more kernels than RBF (Matern).

  Final: Thouroughly tested on real-world data, potentially large data. Compare to multi-task learning using single data covariance kernel.