1. An overview of your BBO approach

My goal in the BBO capstone project is to find the maximum value of eight black box functions. Each week I submit one input for each function, receive the output the following week, and add this to the dataset for next week.

This is an iterative approach. I am using surrogate models such as Gaussian process (GP) and Support Vector Machines (SVM) to estimate the output value using the acquisition functions like UCB and EI. This iterative cycle of modelling->selecting->submitting->updating helps gradually move towards the best performing regions of each function.

2. How your strategy has evolved

My approach has changed from early rounds. Initially, I relied on GP with UCB across all functions, but as more data accumulated, I noticed that some functions responded better to SVM based surrogates, especially on higher dimension functions.

These changes were driven by data trends and model performance. Now my decisions are guided by heuristics such as – switch models when predictions become inconsistent., adjust kappa or EI when exploration stalls.

3. Patterns, data and insights

Across the 12 weeks, several strong patterns emerged. For many functions, a single variable tended to drive most of the variation, acting like a “principal component” in the search space.

For example, in Functions 1 and 2, the second variable consistently determined whether outputs were high or low, while other variables played smaller roles.

In the higher-dimensional functions, extreme values near the boundaries often produced the best results. These insights helped me understand where the search space was most sensitive, allowing me to focus my sampling on the directions that mattered most and reduce attention to areas that consistently produced flat or low-value outputs.

4. Decision-making and iteration

I balance exploration and exploitation by adjusting acquisition parameters: higher kappa or EI encourages exploration when I need to map the space, while lower values push exploitation when I’m close to a promising region. One example that worked well was switching to SVM for Functions 6–8, which stabilised predictions and improved performance after GP became unreliable.

A decision that didn’t work was pushing C (regularisation hyperparameter) too high in SVM, which caused overfitting and inconsistent results. When outcomes don’t match expectations, I adapt by revisiting the surrogate choice, tuning hyperparameters, or widening the search again to avoid getting stuck in local optima.

5. Next steps and reflection

My next steps include testing hybrid strategies—such as alternating between GP and SVM based on uncertainty—and experimenting with adaptive acquisition parameters that change automatically as the search progresses.

My project connects to the wider machine-learning world because it uses the same ideas behind Bayesian optimisation, surrogate modelling and iterative learning. These techniques are used in real ML systems to tune hyperparameters, guide experiments and make decisions when data is expensive or limited. In that sense, my weekly optimisation process is a small-scale version of how ML models learn and improve in practice.

To communicate my results to a non-technical audience, I would communicate my results by focusing on the big picture rather than the technical details. I'd explain that each week I tested new inputs, learned from the outcomes and gradually moved closer to the best possible result, similar to improving a recipe by trying small changes. This helps a non-technical audience understand the progress and value without needing to know the maths behind it.