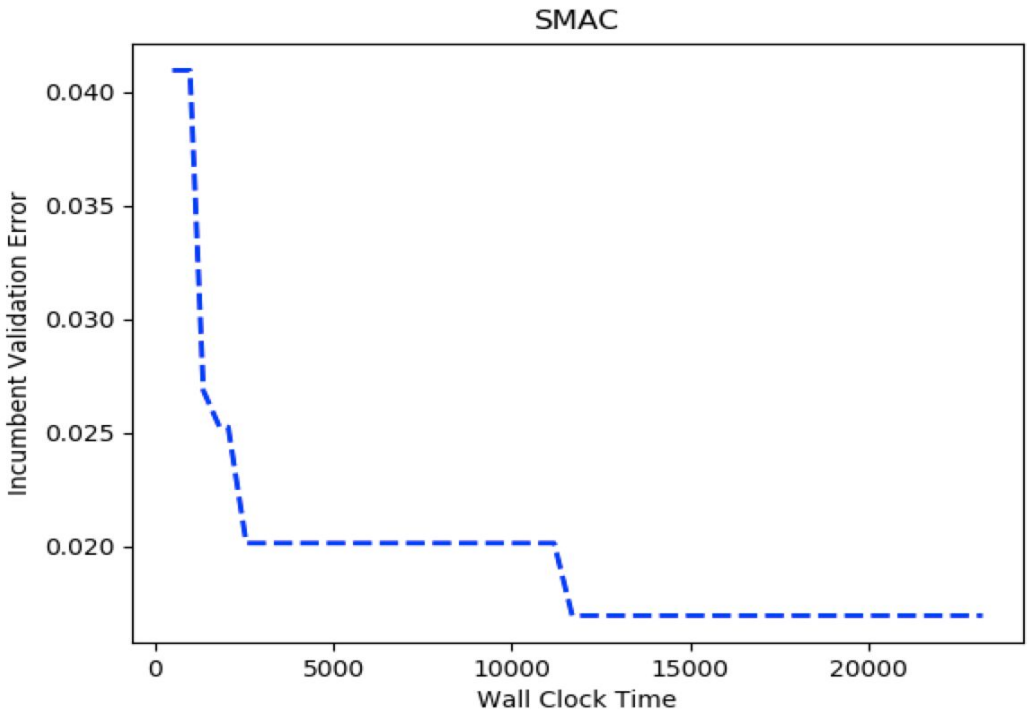
Hyperparameter Optimization - 2
Salih Hasan Siddiqi

This exercise was about optimization of hyperparameters and architectural choices of a fully connected neural network for MNIST. Just like previous exercise, instead of optimizing true objective function, I optimized surrogate function to save computation time during the development phase.

Configuration Space

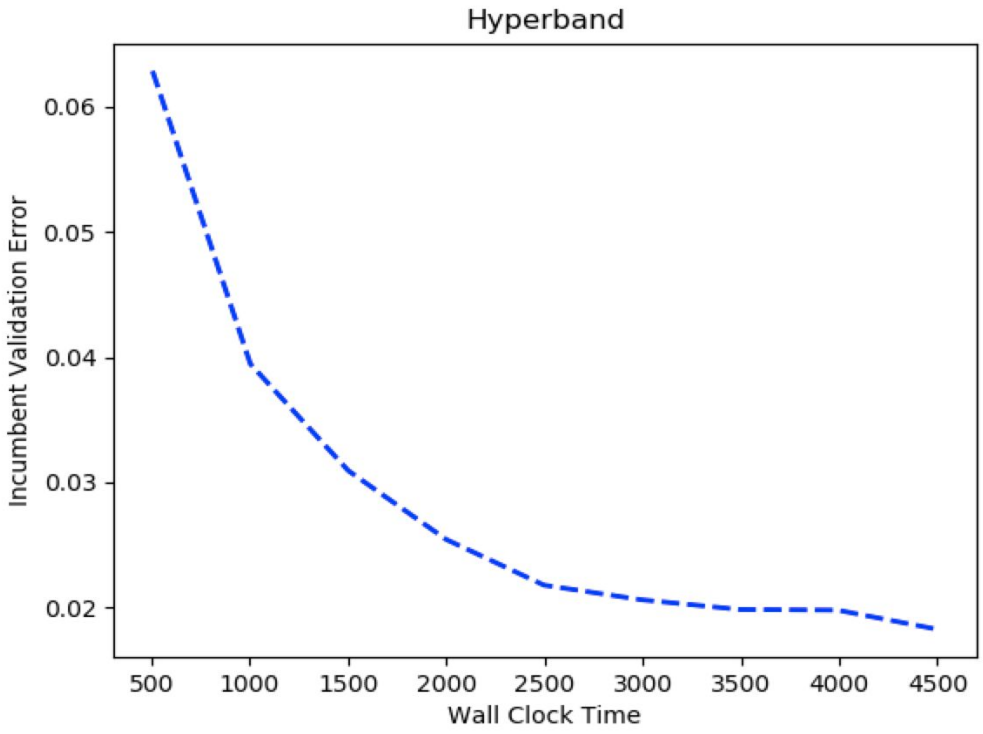
This part of exercise was about implementing configuration space given in the table 1 of exercise sheet. The hyperparameters were from discrete, continuous and categorical configuration space. Moreover, some hyperparameters were conditionally dependent on other hyperparameters. I used ConfigSpace python package to implement configuration space.



In second part, I used SMAC to find good hyperparameter configuration. SMAC uses Bayesian optimization in combination with a simple racing mechanism and employs Random Forest to model objective function. Below is the graph with Incumbent Validation Error on y-axis and wall-clock time on x-axis.

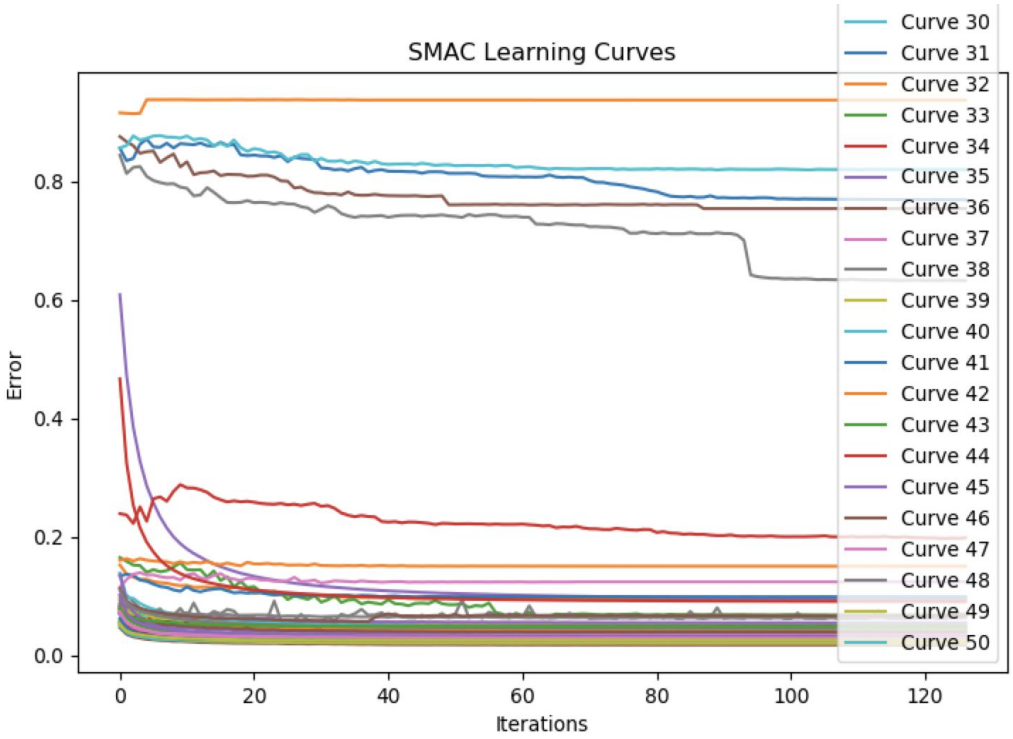
In the third part, I used HpBandSter which is implementation of Hyperband. The underlying principle of the procedure exploits the intuition that if a hyperparameter configuration is destined to be the best after a large number of iterations, it is more likely than not to perform in the top

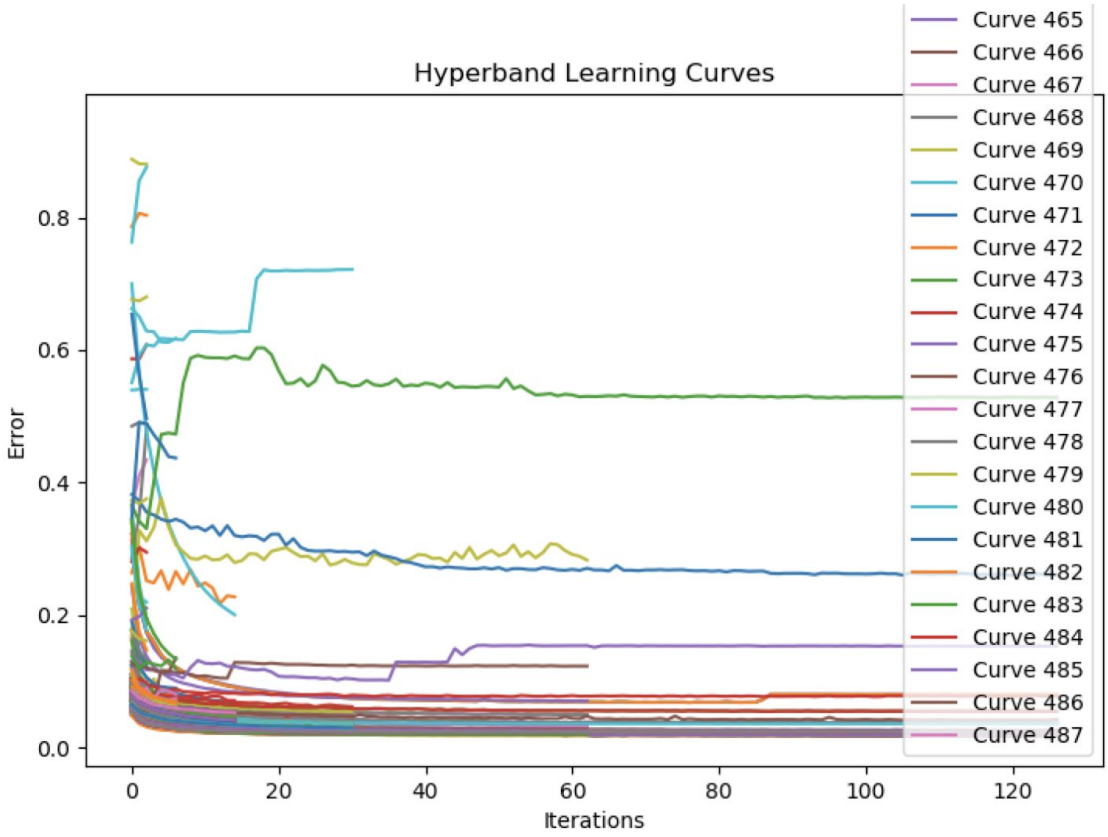
half of configurations after a small number of iterations.



Comparing it to SMAC, Hyperband takes much less time to reach minimum possible error value. This is because of early stopping mechanism Hyperband employs.

Learning Curve





In both graphs, the new learning curves start from lower error value than the previous one. The reason being the new hyperparameter configurations are spit out by SMAC/Hyperband by evaluating previous configurations, which will steer our search for optimized hyperparameters in lower incumbent error space. Furthermore, for SMAC, we have 50 incumbent trajectories while for Hyperband we have 487. So, for Hyperband, we were able to try out much more hyperparameter configurations due to early stopping of learning curves which resulted in early convergence.