Deep Learning Lab

Konstantina Galani
Exercise 5

In this exercise we use random search and Bayesian optimization to find a good hyperparameter setting X for a 3 layer convolutional neural network on CIFAR 10. Instead of an objective function we're using a surrogate to evaluate our regression model. The hyperparameters of the convolutional neural network and their lower and upper bounds are: learning rate = $[10^{-6}, 10^{0}]$, batch size = {32, 512} and number of filters for each layer in $\{2^{4}, 2^{10}\}$.

In Figure 1 we can see the performance of the incumbent after each iteration in random search, where we have 50 iterations.

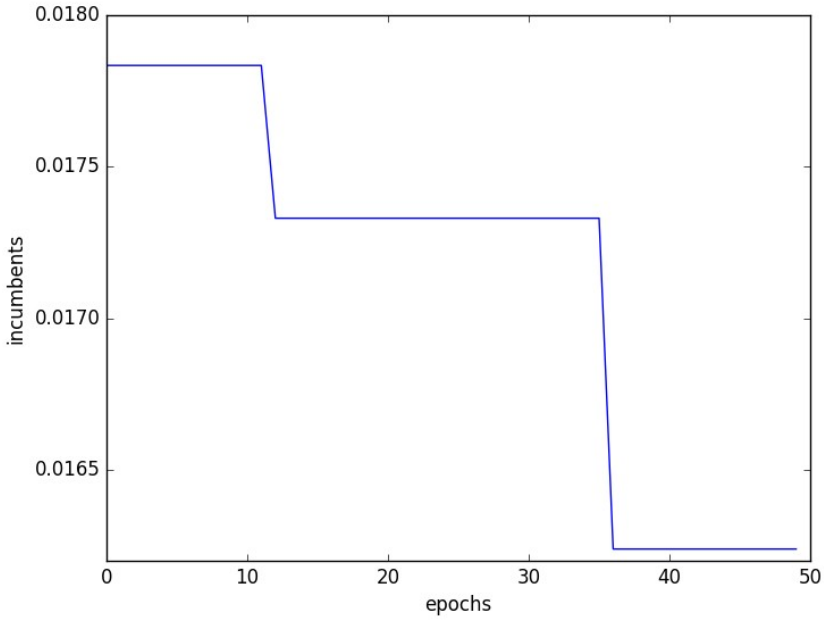

Figure 1
Random search

In Figure 2 we have the plot of the mean performance of the incumbents in random search and bayesian optimization. As we can see after the 3th epoch bayesian optimization reaches a better minimum than random search. Bayesian optimization is not better from the beginning because the first randomly chosen configurations reached a better minimum, but the bayesian takes smarter decisions after each epoch so it improves.

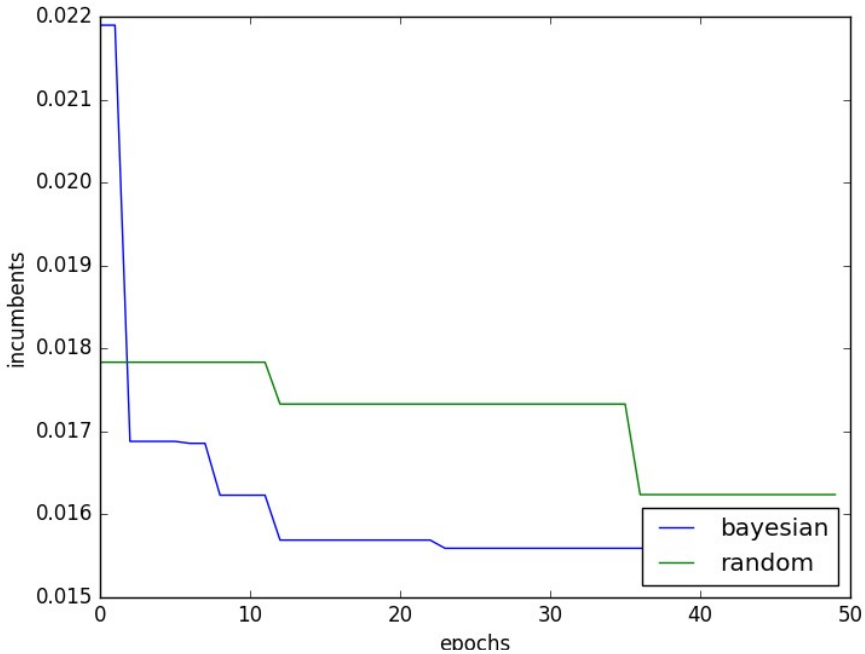

Figure 2
Random search vs bayesian optimization

In Figure 3 we have the cumulative runtime after each iteration for both methods.
From the last 2 plots we can see that Bayesian hyperparameter optimization takes less training steps in order to achieve a comparable result to random search.
The runtime would have been much more if we would have used the true objective function.

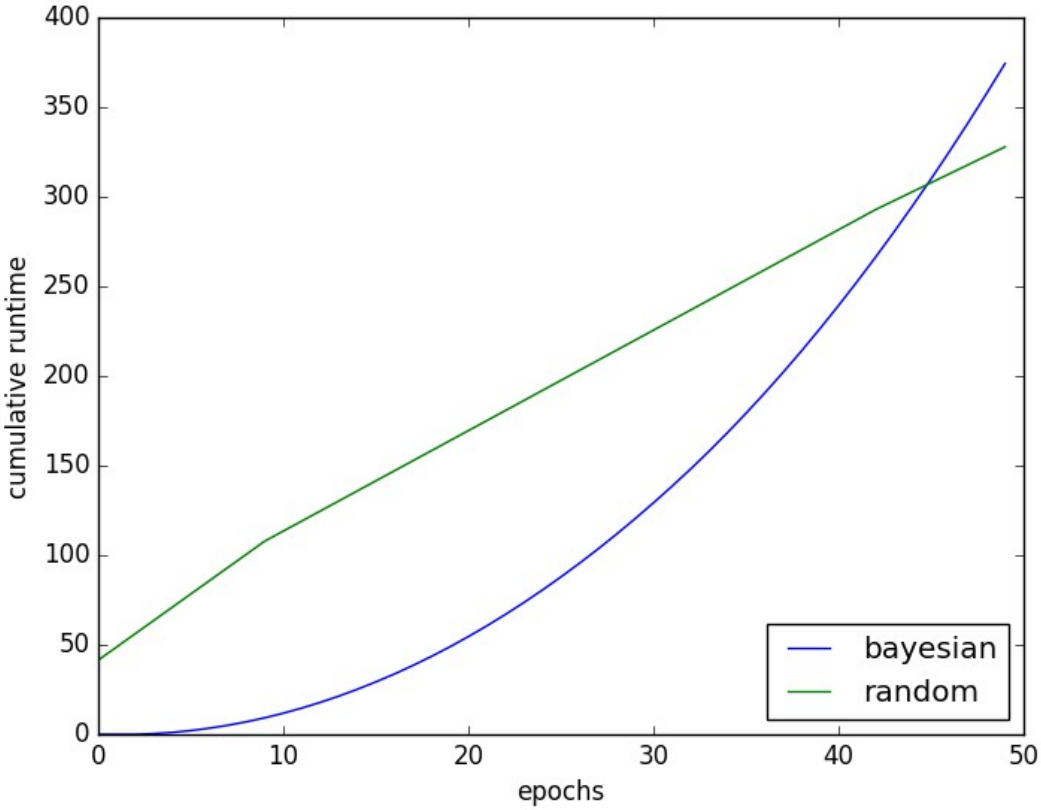

Figure 3
Random search vs bayesian optimization