Lê Jérémy

Practical Bayesian Optimization of Machine Learning Algorithms

Auteur :
Jasper Snock, Hugo Larochelle, Ryan P. Adams

Les algorithmes de Machine Learning sont quasi tous paramétrables. Que ce soit du choix du pas d'apprentissage au choix du nombres d'itérations, ces paramètres peuvent devenir problématique et il devient attrayant de développer des algorithmes « *parameter free* » ou avec peu de paramètres. Cette article explore une méthode appelé « Bayesian Optimization » pour rendre ce choix de paramètre automatique.

L'optimisation Bayésienne considère que les fonctions inconnues à optimiser sont échantillonnés à partir d'un processus gaussien. il faut prendre en considération deux caractéristiques propres à notre problème d'optimisation d'algorithme de Machine Learning:

- Le temps/coût d'apprentissage de l'algorithme qui peut varier énormément en changeant un paramètres (un réseau de neurones avec 10 nœuds cachés coûte plus cher à apprendre qu'un réseau à 1000 nœuds cachés)
- La possibilité d'effectuer des apprentissages en parallèles

Les processus Gaussien sont utiles pour établir et comprendre la sensibilité de leur modèle par rapport à leur hyper-paramètres. Bergsta et al ont démontré que les stratégies *grid search* sont moins efficaces que les stratégies *random search* et ont proposé le « Tree Parzen Algorithm ».

Lorsque l'on fait une optimisation Bayésienne, il est nécessaire de faire deux choix primordial:

- sélectionner un *prior over functions*. Ici, on choisit pour des raisons de flexibilité et de malléabilité « the Gaussian process prior ».
- choisir une fonction d'acquisition qui permet de déterminer le prochain point à évaluer.

Le processus Gaussien est défini par la propriété suivante :

« Tout ensemble fini de N points $\{X_n\}$ $[n=1..N]$ induit une distribution Gaussienne multivariante sur R^n. »

Dans notre cas, X_n correspond à la valeur de la fonction $f(X_n)$.

Il existe plusieurs fonctions d'acquisition pour l'optimisation Bayésienne : Probability of Improvemet, Expected Improvement, GP Upper Confidence Bound. Cet article se concentre sur l' expected Improvement (EI).

Aujourd'hui, l'optimisation Bayesienne d'hyperparamètres n'est pas une technique répandu pour trois raisons :

- Il est difficile de choisir la fonction de covariance et ces hyperparamètres associés.
- l'évaluation de la fonction necessite enormément réduisant la vitesse de l'optimisation voulue.

Lê Jérémy

- Les algorithmes d'optimisation doivent s'adapter au monde et être parraléllisable.

Cependant, l'auteur propose des solutions à chacun de ces problèmes :

- Au lieu d'utiliser l'ARD, squared exponential kernel, il utilise le ARD Matern 5/2 kernel.
- Comme le temps d'execution peut varier énormément d'une régions de l'espace des paramètres à un autre, il propose d'utiliser the expected improvement per second (point that are good and can be evaluated quickly)
- The third point stays unclear… (parallelism)

Il fait ensuite une étude comparative entre son choix d'algorithme à 4 problèmes de Machine Learning (Logistic Regression, Online Latent Dirichlet Allocation, Motif Finding with Structured Support Vector Machines, Convolutional Networks on CIFAR-10). En conclusion, l'efficacité de son approche est prouvé et cette optimisation Bayésienne acquière les hyper-paramètres plus rapidement que les approches des autres auteurs et surpasse l'humain dans le choix des hyper-paramètres sur la compétition CIFAR-10 de 3 %.