View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Ghent University Academic Bibliography

Feature and Model Type Selection using Multi-Objective Optimization for AutoML

Joachim van der Herten JOACHIM.VANDERHERTEN@UGENT.BE
Tom Dhaene TOM.DHAENE@UGENT.BE
Ghent University, Technologiepark 15, 9052, Ghent, Belgium

Keywords: AutoML, Feature Selection, Hyperparameter optimization

Abstract

Automatic identification of relevant features, appropriate model types and their optimal parameters are three fundamental concepts of AutoML. We present a multi-objective method for performing these three tasks, and illustrate its performance on a 30 dimensional data set.

1. Introduction

Recently the use of automated data preprocessing, feature selection, model type selection and result analysis (in short: *AutoML*) have gained a lot of attention. These tasks are often difficult to perform by non-experts, and by enabling systems to perform these steps autonomously results in more off-the-shelf machine learning. This has been crucial for the implementation of machine learning methodology in cloud services.

We present an integrated approach (van der Herten et al., 2016) which builds on earlier proposed model type selection approach by (Gorissen et al., 2009) and is related to Symbolic Regression (Vladislavleva et al., 2008). The approach handles feature selection and model type selection as a first problem, and considers these aspects as a multi-objective optimization problem with both the model accuracy and *complexity* (i.e., the number of input features) as objectives. To obtain the accuracy of a candidate model, its hyperparameters are optimized separately. This allows specific optimization strategies for each model type. An illustration of the approach is given in Figure 1 for three different model types.

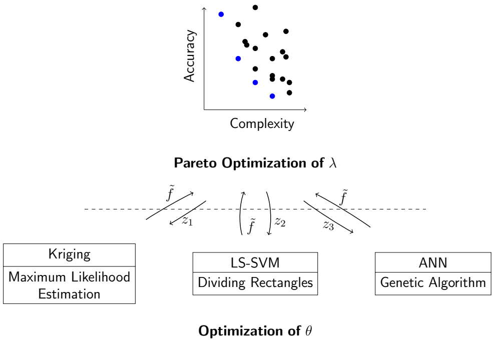

Figure 1. Overview of the two-layered AutoML approach for identification of relevant features, model type and hyperparameters.

2. Approach

Given an unknown function $f : \Omega \rightarrow \mathbb{C}^p$ defined over the input space $\Omega \subset \mathbb{R}^d$. For a distinct set of of query points $X = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_N\}$ the corresponding output labels (classification) or values (regression) $Y = \{f(\boldsymbol{x}_1), ..., f(\boldsymbol{x}_N)\}$ have been observed. Our aim is finding a suitable function $\tilde{f}_{\lambda,\theta} \in S^\star$ from the approximation space. The approximation function is parametrized by its hyperparameters (e.g., regularization parameter) θ and $\lambda = (t, z)$, which consists of the model type t and the included input features represented by z. The subspace of Ω spanned by the included features in z is denoted as $\Omega^\star$, and D_z represents the projection of D into this subspace. The corresponding approximation space of all functions $\tilde{f}_{\lambda,\theta} : \Omega^\star \rightarrow \mathbb{C}^p$ is denoted by $S^\star$.

The choice of $\tilde{f}$ is guided by a pre-set criterion. Given a function $\epsilon \in E$, the set of error functions, and τ a target error specified upfront. The multi-objective

Appearing in *Proceedings of Benelearn 2016*. Copyright 2016 by the author(s)/owner(s).

quality estimator $\tilde{\Lambda}$ is defined as follows:

$$\begin{aligned}\tilde{\Lambda}: \quad E \times \mathcal{S}^\star \times 2^{\Omega^\star} &\to (\mathbb{R}^+, \mathbb{R}^+) \\ (\epsilon, \tilde{f}_{\lambda,\theta}, D_z) &\mapsto \Big(\Lambda(\epsilon, \tilde{f}_{\lambda,\theta}, D_z), \mathcal{C}(\tilde{f}_{\lambda,\theta})\Big). \quad (1)\end{aligned}$$

The first objective is a model quality estimator Λ which is used to assess the accuracy and generalisation performance of $\tilde{f}$: a popular choice is crossvalidation. The second complexity objective $\mathcal{C}$ can be defined as the number of included features in z, but can also be include model type dependant to discourage methods which are computationally expensive to train on high-dimensional spaces.

Multi-objective optimization of Equation 1 over λ with an optimization method such as NSGA-II (Deb et al., 2002) yields a pareto front P. Each individual is optimized with a model-specific strategy to determine an optimal θ, with respect to model accuracy. The model complexity remains constant during the hyperparameter optimization, hence it is not considered during hyperparameter optimization.

The proposed two-layered approach has some key advantages: because the hyperparameter space Θ is different for each model type. This makes inclusion of θ in the multi-objective optimization challenging as it makes implementing crossover and mutation operations less straightforward. Furthermore, this would significantly enlarge the search space and can result in good combinations of features resulting in poor model accuracy if the choice of θ is bad. Finally, we find no evidence in literature that a different choice of model hyperparameters influences the underlying input parameter relevance.

3. Illustration

The 30-dimensional function defined in (Morris et al., 2006) is chosen as illustration using 10 included features $x_1, ...x_{10}$, and was sampled using the FLOLA-Voronoi sampling method (van der Herten et al., 2015). Model types included were SVM (Chang & Lin, 2011) and LS-SVM (Suykens et al., 2002) optimized with the DIRECT algorithm (Jones et al., 1993), Kriging with Maximum Likelihood Estimation (Couckuyt et al., 2014) and ELM (Huang et al., 2011) with hidden layer size and regularization constrant optimized with simulated annealing. Λ for model quality estimation during the hyperparameter optimization of each individual (with varying model type) as well in the multi-objective optimization was chosen as 5-fold crossvalidation with the popular Root Relative Square Error function to account for the generalization error. The complexity objective was set to the number of input parameters included in each model, and multiplied by a factor 2 for kriging and ELM models (as these are more computationally demanding to train for this problem).

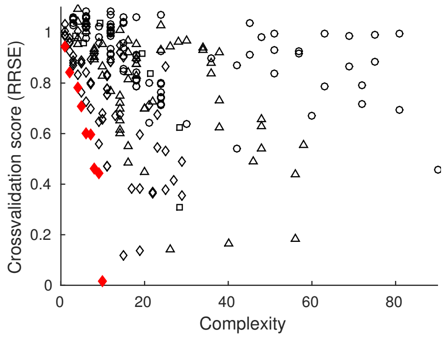

Figure 2. All trained models as part of the adaptive modeling step. Model types are denoted by ◇ for LS-SVM, □ for SVM, △ for Kriging and ○ for ELM models. The pareto front P has been marked in red.

The resulting pareto front, obtained with NSGA-II using 10 generations of 15 individuals is shown in Figure 3. Clearly LS-SVM models are working best for this example. It can also be observed how models with less than 10 parameters automatically underfit, whereas models with additional complexity are not chosen frequently. A model with very good crossvalidation score has also been identified: as it includes exactly $x_1, ...x_{10}$, it manages to fit D very well. As the size of the search space equals 5×2^{30}, it is very unlikely random search would find this solution given only 150 evaluations. In addition, the multi-objective optimization algorithm presents the trade-off which is favourable for real-world applications.

4. Conclusion

A novel AutoML approach for selecting relevant features, appropriate model types and optimizing model hyperparameters automatically was presented and illustrated on a 30-dimensional data set. The method correctly identified the appropriate features, and several model types competed to obtain optimal accuracy.

The downside of the proposed approach is a high computational cost, as each individual of the Genetic Algorithm involves a hyperparameter optimization step. Future work includes incorporating more efficient methods for the multi-objective optimization step (i.e., SMAC or Bayesian optimization), to reduce the number of trained models. In addition we also investigate the performance of the method on real-world high-dimensional engineering applications.

References

Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *2*, 27.

Couckuyt, I., Dhaene, T., & Demeester, P. (2014). ooDACE Toolbox: A Flexible Object-Oriented Kriging Implementation. *Journal of Machine Learning Research*, *15*, 3183–3186.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, *6*, 182–197.

Gorissen, D., Dhaene, T., & De Turck, F. (2009). Evolutionary Model Type Selection for Global Surrogate Modeling. *Journal of Machine Learning Research*, *10*, 2039–2078.

Huang, G.-B., Wang, D. H., & Lan, Y. (2011). Extreme learning machines: a survey. *International Journal of Machine Learning and Cybernetics*, *2*, 107–122.

Jones, D. R., Perttunen, C. D., & Stuckman, B. E. (1993). Lipschitzian optimization without the Lipschitz constant. *Journal of Optimization Theory and Applications*, *79*, 157–181.

Morris, M. D., Moore, L. M., & McKay, M. D. (2006). Sampling plans based on balanced incomplete block designs for evaluating the importance of computer model inputs. *Journal of Statistical Planning and Inference*, *136*, 3203–3220.

Suykens, J. A., Van Gestel, T., De Brabanter, J., De Moor, B., Vandewalle, J., Suykens, J., & Van Gestel, T. (2002). *Least squares support vector machines*, vol. 4. World Scientific.

van der Herten, J., Couckuyt, I., Deschrijver, D., & Dhaene, T. (2015). A Fuzzy Hybrid Sequential Design Strategy for Global Surrogate Modeling of High-Dimensional Computer Experiments. *SIAM Journal on Scientific Computing*, *37*, A1020–A1039.

van der Herten, J., Couckuyt, I., Deschrijver, D., & Dhaene, T. (2016). Multi-objective variable subset selection using heterogeneous surrogate modeling and Sequential Design. *IEEE Congress on Evolutionary Computation (CEC) 2016* (p. To appear). IEEE.

Vladislavleva, E. Y., et al. (2008). *Model-based problem solving through symbolic regression via pareto genetic programming*. Doctoral dissertation.