Working Title

This manuscript (permalink) was automatically generated from trang1618/tpot-ds-ms@97c34e5 on November 2, 2018.

Authors

- **Trang T. Le**
  0000-0003-3737-6565 · trang1618 · trang1618

  Department of Biostatistics, Epidemiology and Informatics, Institute for Biomedical Informatics, University of Pennsylvania, Philadelphia, PA 19104

- **Weixuan Fu**
  0000-0002-6434-5468 · weixuanfu · weixuanfu

  Department of Biostatistics, Epidemiology and Informatics, Institute for Biomedical Informatics, University of Pennsylvania, Philadelphia, PA 19104

- **Jason H. Moore**
  0000-0002-5015-1099 · EpistasisLab · moorejh

  Department of Biostatistics, Epidemiology and Informatics, Institute for Biomedical Informatics, University of Pennsylvania, Philadelphia, PA 19104 · Funded by National Institute of Health Grant Nos. LM010098, LM012601

Abstract

Introduction

For many bioinformatics problems of classifying individuals into clinical categories from high-dimensional biological data, choosing a classifier is merely one step of the arduous process that leads to predictions. To detect patterns among features (*e.g.*, clinical variables) and their associations with the outcome (*e.g.*, clinical diagnosis), a data scientist typically has to design and test different complex machine learning frameworks that consist of data exploration, feature engineering, model selection and prediction. Automated machine learning (AutoML) systems were developed to automate this challenging and time-consuming process. These intelligent systems increase the accessibility and scalability of various machine learning applications by efficiently solving an optimization problem to discover pipelines that yield satisfactory outcomes, such as prediction accuracy. Consequently, AutoML allows data scientists to focus their effort in applying their expertise in other important research components such as developing meaningful hypotheses or communicating the results.

[other AutoML systems]

Tree-based Pipeline Optimization Tool (TPOT) is a genetic programming-based AutoML system that automates the laborious process of designing a machine learning pipeline to solve a supervised learning problem. At its core, TPOT uses genetic programming (GP) [1] to optimize a series of feature preprocessors and machine learning models with the objective of maximizing classification accuracy. While most AutoML systems primarily focus on model selection and hyperparameter optimization, TPOT also pays attention to feature selection and feature engineering in building a complete pipeline. Applying GP with the NSGA-II Pareto optimization [2], TPOT optimizes the accuracy achieved by the pipeline while accounting for its complexity. Specifically, to automatically generate and optimize these machine learning pipelines, TPOT utilizes the Python package DEAP [3] to implement the GP algorithm.

Given no a priori knowledge about the problem, TPOT has been showed to frequently outperform standard machine learning analyses [4,5]. Effort has been made to specialize TPOT for human genetics research, which results in a useful extended version of TPOT, TPOT-MDR, that features Multifactor Dimensionality Reduction and an Expert Knowledge Filter [6]. However, at the current stage, TPOT still requires great computational expense to analyze large datasets such as in genome-wide association studies or gene expression analyses. Consequently, application of TPOT on real-world datasets has been limited to small sets of features [7].

In this work, we introduce two new features implemented in TPOT that helps increase the system's scalability. First, the Dataset Selector (DS) allows the users to pass specific subsets of the

features, reducing the computational expense of TPOT at the beginning of each pipeline to only evaluate on a smaller subset of data rather than the entire dataset. Consequently, DS makes TPOT applicable on large data sets by slicing the data into smaller sets of features (*e.g.* genes) and allowing genetic algorithm to select the best subset in the final pipeline. Second, Template enables the option for strongly typed GP, a method to enforce type constraints in genetic programming. By letting users specify a desired structure of the resulting machine learning pipeline, Template helps reduce TPOT computation time and potentially provide more interpretable results.

Methods

Dataset Selector

TPOT's current operators include sets of feature pre-processors, feature transformers, feature selection techniques, and supervised classifiers and regressions. In this study, we introduce a new operator called Dataset Selector (DS) that enables biologically guided group-level feature selection. Specifically, taking place at the very first stage of the pipeline, DS passes only a specific subset of the features onwards. Hence, with DS, users can specify subsets of features of interest to reduce the feature space's dimension at pipeline initialization. From predefined subsets of features, the DS operator allows TPOT to select the best subset that maximize average accuracy in k-fold cross validation (5-fold by default).

For example, in a gene expression analysis of major depressive disorder, a neuroscientist can specify collections of genes in pathways of interest and identify the important collection that helps predict the depression severity. Similarly, in a genome-wide association study of breast cancer, an analyst may assign variants in the data to different subsets of potentially related variants and detect the subset associated with the breast cancer diagnosis. In general, the DS operator allows for compartmentalization the feature space to smaller subsets based on *a priori* expert knowledge about the biomedical dataset. From here, TPOT selects the most relevant group of features, which can be utilized to motivate further analysis on that small group of features in biomedical research.

Template

Parallel with the establishment of the Dataset Selector operator, we now offer TPOT users the option to define a Template that provides a way to specify a desired structure for the resulting machine learning pipeline (*e.g.* Dataset selector → Feature transform → Decision Trees). Specifying a Template will reduce TPOT computation time and potentially provide more interpretable results.

Current implementation of Template supports linear pipelines, or path graphs, which are trees with two nodes (operators) of vertex degree 1, and the other $n - 2$ nodes of vertex degree 2. Further,

Template takes advantage of the strongly typed genetic programming framework that enforces data-type constraints [8] and imposes type-based restrictions on which element (*i.e.*, operator) can be chosen at each node. In other words, with a Template defined, each node in the tree pipeline is assigned one of the major operator types: dataset selector, feature selection, feature transform, classifier or regressor.

[]

Datasets

We apply TPOT with the new DS operator on both simulated datasets and a real world RNA-Seq gene expression dataset. With both real-world and simulated data, we hope to acquire a comprehensive view of the strengths and limitations of TPOT in the next generation sequencing domain.

Real-world RNA-Seq expression data

We employed TPOT-DS on an RNA-Seq expression dataset of 78 individuals with major depressive disorder (MDD) and 79 healthy controls (HC) from Ref. [9]. Gene expression levels were quantified from reads of 19,968 annotated protein-coding genes and underwent a series of preprocessing steps including low read-count and outlier removal, technical and batch effect adjustment, and coefficient of variation filtering. Consequently, whole blood RNA-Seq measurements of 5,912 genes were obtained and are now used in the current study to test for association with MDD status. We use the 23 subsets of interconnected genes called modules identified from the RNA-Seq gene network module analysis [9] as input for the DS operator.

Simulation methods

The simulated datasets were generated using the R package privateEC, which was designed to simulate realistic effects to be expected in gene expression or resting-state fMRI data. In the current study, to be consistent with the real expression dataset, we simulate interaction effect data with $m = 200$ individuals (100 cases and 100 controls) and $p = 5,000$ real-valued features with 4% functional (true positive association with outcome). Briefly, the privateEC simulation induces a differential co-expression network random normal expression levels and permute the values of targeted features within the cases to generate interactions. Full details of the simulation approach can be found in Refs. [10,9]. Further, by imposing a large number of background features (no association with outcome), we seek to assess TPOT-DS's performance in accommodating large numbers of non-predictive features.

Evaluation metrics

[...]

Results

Discussion

References

1. https://dl.acm.org/citation.cfm?id=280485

2. **A fast and elitist multiobjective genetic algorithm: NSGA-II**
K. Deb, A. Pratap, S. Agarwal, T. Meyarivan
IEEE Transactions on Evolutionary Computation (2002-04) https://doi.org/bnw2vv
DOI: 10.1109/4235.996017

3. **DEAP: Evolutionary Algorithms Made Easy**
Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, Christian Gagné
Journal of Machine Learning Research (2012) http://www.jmlr.org/papers/v13/fortin12a.html

4. **Evaluation of a Tree-based Pipeline Optimization Tool for Automating Data Science**
Randal S. Olson, Nathan Bartley, Ryan J. Urbanowicz, Jason H. Moore
Proceedings of the 2016 on Genetic and Evolutionary Computation Conference - GECCO '16 (2016) https://doi.org/gfgqv2
DOI: 10.1145/2908812.2908918

5. **Identifying and Harnessing the Building Blocks of Machine Learning Pipelines for Sensible Initialization of a Data Science Automation Tool**
Randal S. Olson, Jason H. Moore
arXiv (2016-07-29) https://arxiv.org/abs/1607.08878v1

6. **Toward the automated analysis of complex diseases in genome-wide association studies using genetic programming**
Andrew Sohn, Randal S. Olson, Jason H. Moore
Proceedings of the Genetic and Evolutionary Computation Conference on - GECCO '17 (2017) https://doi.org/gfgqv3
DOI: 10.1145/3071178.3071212

7. **Integrated machine learning pipeline for aberrant biomarker enrichment (i-mAB): characterizing clusters of differentiation within a compendium of systemic lupus erythematosus patients**
Trang T. Le, Nigel O. Blackwood, Jaclyn N. Taroni, Weixuan Fu, Matthew K. Breitenstein
arXiv (2018-03-08) https://arxiv.org/abs/1803.04487v1

8. **Strongly Typed Genetic Programming**
David J. Montana
Evolutionary Computation (1995-06) https://doi.org/ct3mnb
DOI: 10.1162/evco.1995.3.2.199

9. **Identification and replication of RNA-Seq gene network modules associated with depression severity**
Trang T. Le, Jonathan Savitz, Hideo Suzuki, Masaya Misaki, T. Kent Teague, Bill C. White, Julie H. Marino, Graham Wiley, Patrick M. Gaffney, Wayne C. Drevets, … Jerzy Bodurka
Translational Psychiatry (2018-09-05) https://doi.org/gd7jx7
DOI: 10.1038/s41398-018-0234-3 · PMID: 30185774 · PMCID: PMC6125582

10. **Differential co-expression network centrality and machine learning feature selection for identifying susceptibility hubs in networks with scale-free structure**
Caleb A Lareau, Bill C White, Ann L Oberg, Brett A McKinney
BioData Mining (2015-02-03) https://doi.org/gb5fpr
DOI: 10.1186/s13040-015-0040-x · PMID: 25685197 · PMCID: PMC4326454