A Hybrid HSIC-ACO algorithm for Variable Selection in Process Engineering

Shameek Ghosh\(^a\), Srikant Jayaraman\(^b\), Shikha Bhat\(^b\), V. K. Jayaraman\(^{a,c,*}\)

\(^a\)Evolutionary Computing & Image Processing Group, Centre for Development of Advanced Computing, Pune, India
\(^b\)Center for Modeling and Simulation, University of Pune, Pune, India
\(^c\)Shiv Nadar University, Tehsil Dadri, District Gautam Budh Nagar - 203207, Uttar Pradesh, India
\(^*\)Corresponding author

Abstract

Recently, data mining and machine learning techniques have been increasingly applied in process engineering. Various successful applications include fault detection and development of data driven models. While fault detection is useful for steady operation of the plant, data driven models can be employed for robust prediction of structure activity relationships. Many of these models require nonlinear classification techniques. The success of these techniques relies on the integration of informative domain knowledge to the concerned methods. In this study, we propose a hybrid Ant Colony optimization (ACO) based variable selection approach in conjunction with Support Vector Machines (SVM) to determine informative subsets of process variables that may help detect faults efficiently, making the fault detection model more robust in the process. In addition, we employ a Hilbert Schmidt Independence Criterion (HSIC) based variable ranking heuristic to guide ACO towards better search spaces. Performance testing of HSIC-ACO was carried out on the benchmark Tennessee Eastman Process challenge and large scale QSAR prediction data collected from relevant sources. Our results demonstrate improved fault detection and structure-activity prediction capabilities using the HSIC-ACO algorithm.

Keywords: Fault Detection, Ant Colony optimization, Hilbert Schmidt Independence Criterion.

1. Introduction

Efficient process engineering techniques focus on the optimization of design, operation, and control parameters of various chemical, physical, and biological processes through the aid of supervisory computer-based control methods (Downs, Vogel, 1993). For computational analysis, usually large sets of sensor measurements may be collected over time by a process monitoring system. These measurements are composed of a large set of variables that influence the state of the process at a given timestamp. Depending on the type of study undertaken, the state changes are recorded against the concerned set of variables. Once the required data has been collected, statistical models built using such data may uncover numerous insights to the given problem. A major limitation in this context comes from the complexity of the process data. Constructing prediction models from such data thus turns out to be very tedious and the resulting model is normally quite inferior due to inclusion of irrelevant and redundant variables in the
model. In machine learning parlance this is popularly known as the variable selection problem (Guyon, Elisseeff, 2003), where an optimal subset of the most relevant variables is selected for building robust learning models. Variable selection may thus help to alleviate the curse of dimensionality, speed up the learning process, and enhance the model’s generalization capability and interpretability. These may be broadly categorized as: Wrappers and Filters (Gupta et al, 2006; Nikumbh et al, 2012). Wrappers employ a learning algorithm which attempt to determine an optimal subset of suitable variables for goodness of fit to the problem, using the algorithm’s learning mechanism. In contrast, filters evaluate the ability of a variable based on the inherent data characteristics using statistical techniques and may report a ranking of the concerned.

In this work, we present the Hilbert-Schmidt Independence criterion (HSIC) based variable ranking and a hybrid Ant Colony Optimization (ACO) based wrapper in conjunction with Support Vector Machines (SVM) as a filter-wrapper algorithm for variable subset selection and simultaneous detection of process faults in the Tennessee Eastman Challenge Process data (Downs, Vogel, 1993; Lyman, Georgakis, 1995) and QSAR modeling data for estimating molecular activity levels in biological processes (http://www.coepra.org/). The HSIC (Song et al, 2007) provides a statistical measure to determine the dependence between a set of process variables (which makes up a sample entry) and the related system state (fault or molecular activity in this case). The HSIC criterion is later used with the backward elimination strategy (BAHSIC) to obtain a ranking of all variables based on the HSIC measure. In the context of our study, BAHSIC (Song et al, 2007) is treated as the filter which is fed as an input to the ACO-SVM wrapper algorithm for simultaneous variable selection and classification as will be described next.

2. Methodology
2.1. Ant Colony Optimization (ACO)

The first ACO algorithms were introduced by Marco Dorigo and colleagues in the early 1990’s (Dorigo, Gambardella, 1997). Ant Colony Optimization (ACO) has been motivated by the real behavior of a colony of ants while foraging for food. While searching for food, ants normally begin by exploring surrounding areas of their nest randomly. While traversing, ants deposit chemical pheromones on the tracks they cover. Pheromone smell is also a form of implicit communication between ants. Owing to this, ants choose paths probabilistically, with a strong concentration of pheromone deposition.

According to ACO’s application to the Travelling Salesman Problem (TSP) problem (Dorigo, Gambardella, 1997), we consider a complete graph \( G = (V, E) \) where \( V = \{v_1, v_2, ..., v_n\} \) and \( E = \{e_1, e_2, ..., e_k\} \). The TSP objective is to find a circuit in \( G \) that contains each vertex exactly once and whose length is minimal. So the search space consists of all possible tours in \( G \).

As per ACO for TSP, a random vertex may be selected as the starting node. All ants are placed at this node. An ant selects the next node, based on the probability transition given by equation (1).

\[
P(e_j) = \frac{\tau_i^\alpha \cdot \eta_j^\beta}{\sum_{j \text{unvisited nodes}} \tau_i^\alpha \cdot \eta_j^\beta}
\]  

(1)
A Hybrid HSIC-ACO algorithm for Variable Selection in Process Engineering

where $\eta$ is a heuristic value associated with each feasible solution ($e_i$) component and $\tau$ is the pheromone concentration on an edge. Furthermore, the exponents $\alpha$ and $\beta$ are parameters whose values determine the importance assigned to pheromone concentration and heuristic information. This kind of a probabilistic edge selection continues until a single ant builds a complete tour. Subsequent ants follow a similar process to construct their solutions. At the end of an iteration, a global pheromone update is applied to edges of the best tour out of all ants, as given in equation 2.

$$\tau_{r,s} = (1 - \alpha) \cdot \tau_{r,s} + \alpha \Delta \tau_{r,s}$$

$$\Delta \tau_{r,s} = \begin{cases} \tau^{-1} & \text{if } (r, s) \in \text{best}_-\text{tour} \\ 0 & \text{else} \end{cases}$$

Here $\tau_{r,s}$ denotes the pheromone concentration between nodes $r$ and $s$. $\alpha$ is the pheromone decay parameter and $c$ gives the cost of the best tour. Similarly, a local update simulating pheromone evaporation is also performed for the edges not selected as part of the best tour in equation (3) as below.

$$\tau_{r,s} = (1 - \rho) \cdot \tau_{r,s}$$

In (3), $0 < \rho < 1$ is an algorithmic parameter to sustain evaporation of pheromone from edges that are not part of the best tour.

Dorigo et al (1997) later followed up the earlier ACO algorithm with a new version called Ant Colony System (ACS), towards efficiency improvement in the TSP. This process involved the generation of a random value called $q$, between 0 and 1, which was tested against a threshold $q_0$ (user defined). An exploitation, where the best available partial solution component would be chosen (the shortest edge with maximum pheromone concentration for TSP) which is choosing the maximum among $[\tau_i \cdot \eta(e_i)]^{\alpha}$, constituted the next option if $q$ was less than $q_0$. Otherwise, exploration, where a random solution component, according to a probability distribution, would be selected. Such measures thus overcame many problems which normally a greedy algorithm would suffer from, for example the solution search being stuck in local optima.

2.2. Backward Elimination using HSIC

The HSIC criterion is used to measure the dependence between two multivariate random variables. The key idea is that good variables should maximize the dependence between variables and class labels. For computing the HSIC of a subset of variables, we employed the definitions as given by Song et al (2007). Having defined the HSIC criterion we use it with the backward elimination strategy to obtain a ranking of all variables. Backward elimination involves starting with all candidate variables, testing the removal of each variable/s using the HSIC comparison criterion, deleting the variable or a certain percentage of variables, that improves the model the most by being deleted, and repeating this process till no more variables are left. Finally we obtain a ranking of all process variables given by the BAHISC method. In the context of our study, BAHISC is treated as the heuristic filter which is being fed as input to the ACO wrapper algorithm described next.
2.3. HSIC-ACO based Variable Selection

In our ACO based variable selection model, a solution may be represented as a set of variable indices. Thus if a process dataset consists of 50 variables, then a possible solution could be a variable subset comprising of \{10,21,32,57,84\} with subset size as 5. Any variable index could thus be a part of the solution vector, where the vector or subset size is specified by the user.

In correlation to the ACO-TSP model, we thus consider each process variable as a node (in the TSP graph). The ants could thus be randomly placed on a variable, from which they commence their movement. Based on the probabilistic edge selection mechanism using equation 1, an ant may move to another variable. It is during this edge selection process that we introduce the HSIC measure of a variable as the η heuristic component. This procedure of probabilistic variable selection continues till a solution vector of the relevant size is constructed. When a vector of the required size is obtained, a corresponding reduced dataset with the corresponding vector variables is generated. The reduced dataset is then fed as an input to a classifier like SVM (Boser, Guyon, Vapnik, 1992), which subsequently reports back a 10 fold classification cross validation accuracy (10 fold CVA). The 10 fold CVA is thus assigned as the fitness function value for the corresponding ant’s solution vector.

At the end of ACO iteration, the solution vector with the maximum 10 fold CVA is selected as the solution that undergoes a global pheromone update as given in equation (2). Thus if \{6,12,78,101,132\} is the best solution vector in an iteration, then edges represented by 6-12, 12-78, 78-101, 101-132 and 132-6 are updated. All other edges are made to undergo a pheromone evaporation process as given in equation (3).

The above procedure is thus repeated for a maximum number of iterations, after which the ACO may be terminated and the solution vector in the final iteration may be reported as having the most optimal subset of variables.

The extracted set of informative variables may now be used to build a robust fault detection or classification model.

2.4. Support Vector Machines

Support Vector Machines (SVM) fall under a class of learning algorithms based on the statistical learning theory developed by Vapnik (Boser, Guyon, Vapnik, 1992). SVMs are heavily used for purposes of classification and regression. But the quality of a model generated by an SVM is heavily dependent on the input variables that are provided and the internal parameters considered by SVM. SVM employs a hyper plane to divide a set of binary-labeled data, maximizing the margin between the nearest data points of each label, in the process. For linearly non-separable data, SVM scales and transforms the input points into a higher dimensional variable space and then finds a suitable linear hyperplane for separating the data points for classification. For implementation purposes, we have employed the libSVM (Chang, Lin, 2011) software library.

3. Results and Discussion

The HSIC-ACO algorithm has been used to solve the problem of simultaneous variable selection and fault detection in the benchmark Tennessee Eastman Process (TEP) and for large scale feature selection using QSAR (quantitative structure activity relationship) data obtained from the CoEPrA competition datasets (accessible online at: http://www.coepra.org).
For the TEP process fault detection problem, there are 52 variables in the system comprising different pressures, temperatures etc. monitored over a certain period of time. This consists of 21 different types of faults. Among these, we consider two representative fault detection datasets for binary and multi-class prediction problems. For the binary classification problem, we use faults due to step change in D feed temperature and reactor cooling water inlet temperature. In the case of multiple faults, we use three faults due to step changes in reactor cooling water inlet temperature, random variations in D feed temperature and random variations in reactor cooling water inlet temperature. Each of the specific faults in both the datasets comprise of 480 observations respectively. As mentioned before, we also use the CoEPRA1 (comprises 89 samples with 5787 features) and CoEPRA2 (comprises 76 samples with 5144 features) classification data for QSAR modeling using the proposed algorithm. The parameters of the hybrid HSIC-ACO algorithm are as given in Table 1.

Table 1: HSIC-ACO algorithm parameters

<table>
<thead>
<tr>
<th>Algorithmic Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Ants</td>
<td>20</td>
</tr>
<tr>
<td>Number of Iterations</td>
<td>20</td>
</tr>
<tr>
<td>Pheromone Decay Parameter ($\alpha$)</td>
<td>0.98</td>
</tr>
<tr>
<td>SVM: Cost, Gamma, Kernel</td>
<td>32.0, 0.0001220703125, RBF</td>
</tr>
</tbody>
</table>

We carried out extensive simulations using the above tuned parameters to obtain the average results as given Table 2. Based on numerous tests we noticed that our results would generally converge by the end of 20 iterations.

Table 2: 10 fold Cross-Validation Classification Accuracies using HSIC-ACO

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Variable Subset Size</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary Faults (using faults 3 &amp; 4)</td>
<td>4</td>
<td>99.89%</td>
</tr>
<tr>
<td>Multiple Faults (using faults 4, 9 &amp; 11)</td>
<td>10</td>
<td>90.83%</td>
</tr>
<tr>
<td>CoEPRA1 classification data</td>
<td>100</td>
<td>82.02%</td>
</tr>
<tr>
<td>CoEPRA2 classification data</td>
<td>100</td>
<td>88.15%</td>
</tr>
</tbody>
</table>

The reported literature on TEP fault detection has been varied in its assessment of fault detection techniques. There may be many combinations of faults one may consider to test a method, yet specifically earlier methods consider faults 4, 9 and 11 mostly due to their overlapping nature and consequent difficulty in detection. Recent results using PSVM, ISVM, LDA and QDA report accuracies in the range of 90-94% CVA for the multiple faults detection problem (Kulkarni, Jayaraman, Kulkarni, 2005; Verron, Tiplica, Kobi, 2008). In the case of the binary faults detection, we noticed that HSIC-ACO consistently outperformed filter algorithms (like Information gain and BAHSIC). Additionally the CoEPRA1 and CoEPRA2 classification accuracies are comparable to results reported earlier using recent evolutionary methods (Srivastava et al, 2012). Results of HSIC-ACO thus seem encouraging in selecting smaller informative sets of variables that may contribute to robust fault modeling applications in process engineering and biosciences molecular activity detection.
4. Conclusion
The HSIC-ACO algorithm employs both the filter and wrapper methods to obtain relevant subsets of variables that are important for fault detection and QSAR modeling. The BAHSIC heuristic ranking also provides more possibilities for exploring the search space effectively that seem to have helped in the selection of important variables. Finally HSIC-ACO is flexible and robust since we can adapt it to a given problem and related domain constraints.

Acknowledgements: VKJ gratefully acknowledges the Department of Science and Technology (DST), New Delhi, India for financial support. The authors also acknowledge the Centre for Modeling and Simulation, University of Pune and C-DAC, Pune for their kind support.

References