COMB-PSO: Feature Selection of Kinases for Prediction of Neurite Outgrowth

Nathan C. Fox[1], Hassen Dhrif[1], Hassan Al-Ali[2,3,4], Stefan Wuchty[1], John L. Bixby[2,4,5], Vance P. Lemmon[2,4]

1. Department of Computer Science, University of Miami, FL, USA. 2. The Miami Project to Cure Paralysis, University of Miami, Miami, FL, USA. 3. Peggy and Harold Katz Family Drug Discovery Center, University of Miami Miller School of Medicine, Miami, FL, 33136, USA.
4. Center for Computational Science, University of Miami, Miami, FL, USA. 5. Department of Molecular & Cellular Pharmacology, University of Miami Miller School of Medicine, Miami, FL 33136, USA.

1. Introduction

Drug discovery remains a field with extremely high failure rates. Consequently, it is an excellent place to test data analysis and optimization algorithms. Here, we have a dataset of 190 kinases, each of which have been treated in vitro with 256 kinase inhibitors and assayed for activity, resulting in an "inhibition profile". Each of the inhibitors was then tested on mouse hippocampal neurons and assayed for neurite outgrowth. For machine learning classification, the feature data are the inhibition profiles of 190 kinases (features); the classifications are True/False labels for "Induces Neurite Outgrowth." The goal is to find the smallest possible subset of features that best predict an inhibitor's effect on neurite outgrowth. This was done previously by Al-Ali et al. in 2015 using a process called Maximum Relevance-Support Vector Machine (MR-SVM). They took the top 50 kinases as scored by the Maximum Relevance algorithm, then used a Support Vector Machine (SVM) classifier to iteratively prune kinases whose removal did not impact classification performance[2].

Combined Continuous and Binary Particle Swarm Optimization (COMB-PSO) optimizes a function using a swarm of particles to search an n-dimensional space[1]. We hypothesize that it can be used as a feature selection algorithm to find smaller feature subsets that give high classification accuracy, sensitivity, and specificity.

Here, the algorithm's search space has 190 dimensions. Thus, a point in this space has 190 components, each representing the presence or absence of a particular feature. This subset that a point represents can be tested for accuracy, sensitivity, and specificity using a classifier. The algorithm searches this 190-dimensional space for the smallest subset with the highest classification scores.

2. COMB-PSO Algorithm

COMB-PSO uses a virtual swarm of virtual particles that explore the search space as a collective. Each particle has a 190-dimensional position vector, *x*, and a 190-dimensional velocity vector, *v*. These vectors are continuous; x_i represents a probability of inclusion. When a continuous position is tested for classification score, it is converted into a binary vector, *b*, that represents a true feature subset. The equations for position, velocity and conversion to binary are below[1]:

$$(1) \qquad \vec{x}_i(t+1) = \vec{x}_i(t) + \vec{v}_i(t)$$

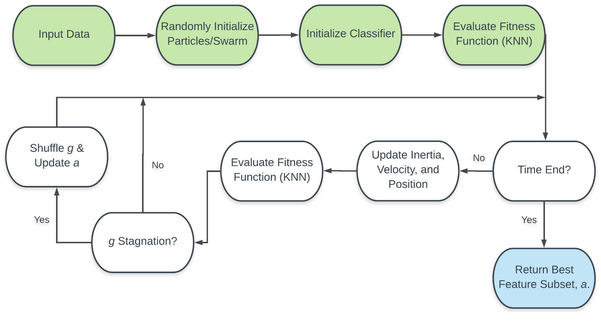
$$(2) \qquad \vec{v}_i(t+1) = w\vec{v}_i(t) + c1\vec{R1}_i(\vec{p}_i - \vec{x}_i(t)) + c2\vec{R2}_i(\vec{g}_i - \vec{x}_i(t)) + c3\vec{R3}_i(\vec{a}_i - \vec{x}_i(t))$$

$$(3) \qquad b_i = \begin{cases} 1, \text{ if } rand() < S(x_i) \\ 0, \text{ otherwise} \end{cases}$$

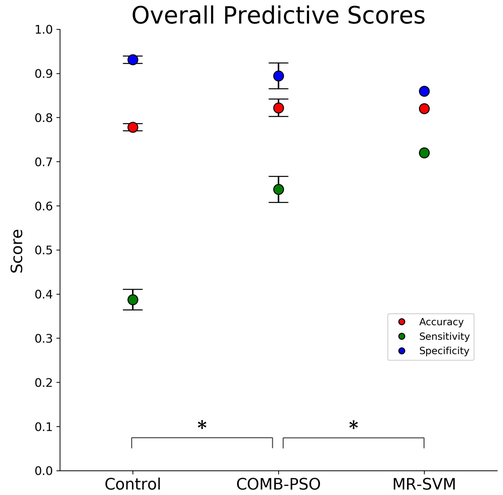
x, *v*, and *b* are position, velocity, and binary position vectors as described above. *w* is a particle's inertia, a value calculated to weight the impact of the current velocity on the next velocity. *c*1, *c*2, and *c*3 are constant weights. *R*1, *R*2, and *R*3 are 190-component randomly generated vectors from a uniform distribution, [0.0, 1.0). *p*, *g*, and *a* are described below. *rand*( ) is a randomly generated scalar from a uniform distribution, [0.0, 1.0). *S*( ) is the logistic transformation[1].

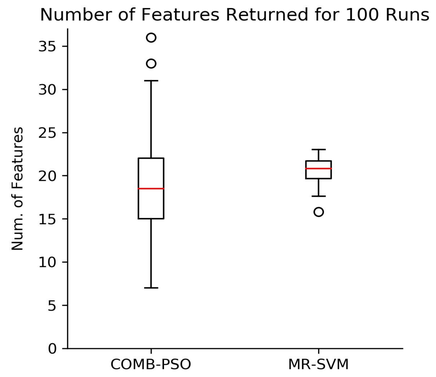
Each particle maintains a record of its personal best position, *p*. Additionally, the swarm maintains a record of the global best position, *g*, and the archived best position, *a*. For each time step, each particle updates its velocity, uses the new velocity to update its position, and evaluates the fitness of the new position. If the new position has greater fitness than *p* or *g*, they are updated. Additionally, COMB-PSO has two behaviors that differ from the original PSO algorithm[1].

1. **Variable Inertia:** Each particle maintains its own inertia, based on current distance from *g*. If the particle is close to *p*, the inertia is lower and vice versa. This allows the particles to "slow down" when near a potential optimum.
2. **Shuffle and Archive:** If *g* stagnates and does not change for 3 time steps, it is saved into *a*, and then randomly reinitialized. This behavior is designed to jolt the swarm out of local optima where it may get prematurely trapped.



3. COMB-PSO Finds a Smaller, More Informative Subset





Distribution of number of features returned in a subset by COMB-PSO for 100 runs and MR-SVM for 10 runs.

Mean classification scores from 10-fold cross-validation (CV) using the subsets returned from 100 runs of COMB-PSO and 10 runs of MR-SVM. Control: All features were used for 100 runs of 10-fold CV. Error Bars not shown for MR-SVM for clarity.

* = $p < 10^{-49}$. Error Bars are one Standard Deviation.
Accuracy = TP+TN/FP+FN. Sensitivity = TP/TP+FN. Specificity = TN/TN+FP.
{TP, FP, TN, FN} = {True Positive, False Positive, True Negative, False Negative}

4. COMB-PSO Returns Fewer Consistently Informative Features

COMB-PSO

Kinase	Frequency of Inclusion
EGFR	0.90
TTK	0.52
SIK2	0.49
ROCK1	0.46
CHUK	0.43
ERBB4	0.39
ROCK2	0.37
MUSK	0.26
GSK3A	0.25
LTK	0.25

MR-SVM[2]

Kinase	Frequency of Inclusion
PIK3CD	1.00
EGFR	0.99
MAPK11	0.86
MAP4K4	0.70
MAPK14	0.67
TTK	0.61
MUSK	0.58
ROCK2	0.56
FLT4	0.53
PDGFRA	0.53

Top 10 kinases by frequency of inclusion in the returned subset. Left: COMB-PSO. Right: MR-SVM[2]. Bolded kinases appear in the top 10 for both algorithms.

5. Methods

Panel 3: The COMB-PSO algorithm was run 100 times with a K-Nearest Neighbors (KNN) classifier and a fitness function that ran a 10-fold cross-validation (CV) and weighted accuracy, sensitivity, and fewer features respectively as 0.3, 0.5, and 0.2. The same classifier was used 100 times in a 10-fold CV on the full 190-feature set as a control. The 100 returned feature sets were used in a fresh 10-fold CV to test accuracy, sensitivity, and specificity for predictive power.

Panel 4: The 100 feature sets returned from the experiment in Panel 4 were analyzed to determine frequency of inclusion for each of the 190 possible features. The top 10 are reported and compared to the top 10 features reported by Al-Ali, et. al. in a previous study with this data. Al-Ali, et. al. used a combination of the Maximum Relevance algorithm to get a top 50, then a greedy selection approach with an SVM to prune it further[2].

The COMB-PSO algorithm was implemented in Python 3.6 using the numpy, sklearn, pandas, and matplotlib scientific computing and visualization libraries. Kinase inhibitor data was given by the Lembix Lab. All computation was carried out on either the University of Miami Center for Computational Sciences Pegasus supercomputer or a Dell Precision 5520 running Ubuntu 16.04 with an Intel® Core™ i7-7820HQ CPU and 16 GB of RAM.

6. Results

The COMB-PSO feature selection algorithm reduced a 190-feature dataset by a factor of 10 and significantly improved both accuracy and sensitivity. COMB-PSO had a mean accuracy of 82.2%, sensitivity of 63.7%, and specificity of 89.5% (Std Dev: 2.0%, 2.9%, 2.9%). The control had a mean accuracy of 77.8% sensitivity of 38.7%, and specificity of 93.1% (Std Dev: 0.8%, 2.3%, 0.8%). Al-Ali, et al. reported a mean accuracy of 81.8%, sensitivity of 74.4%, and specificity of 84.4% (Std Dev: 3.2%, 8.5%, 2.1%). The mean number of features in the returned subsets was 18.97. Al-Ali, et al. reported 20.26. The specificity significantly decreased from 93.1% to 89.5%, but by a relatively small amount compared to the progress made in sensitivity.

COMB-PSO returned 2 features more than half the time, epidermal growth factor receptor (EGFR) and TTK protein kinase (TTK). Excluding those two, COMB-PSO returned 16 specific features more than 20 percent of the time. Four of the top ten most commonly returned features were also found in the top ten most commonly used features reported by Al-Ali, et al.

7. Conclusions

As computing power and available storage space continue to grow, "big data" become increasingly common. Genome sequencing, DNA microarray assays, and drug discovery screens all generate enormous amounts of data that are difficult to organize into a usable form. Despite greater computing power, optimization for efficiency remains a crucial skill for effective handling of large datasets. Part of that process includes reducing large datasets to the minimum necessary subset that remains maximally effective. This is especially true in machine learning, where the "curse of dimensionality" can exponentially increase processing time and require impossible amounts of data to show statistical significance.

Here, we evaluated the use of a new variant of Particle Swarm Optimization (PSO), COMB-PSO, for use in feature selection to reduce a high-dimensional feature set to a more manageable size. The algorithm successfully improved accuracy by 6% and sensitivity by 65% while simultaneously shrinking the required number of features by 90%.

However, COMB-PSO required a somewhat large amount of computing power, 4 cores for approx. 7 hours. Additionally, it only selected 2 features more than 50 percent of the time, implying that there might be redundancy among features or that the algorithm is failing to reduce noise. Finally, evaluative experiments not shown indicated that the algorithm converges extremely quickly. This extreme speed might indicate premature convergence on a local optimum and failure to adequately explore the full search space, despite the addition of the Shuffle and Archive behavior. Algorithm parameters were empirically chosen, relying on benchmark tests and previous studies on PSO parameter selection. In order for the PSO family of algorithms to be useful in a broad scale, further work on quickly determining effective parameters could be important.

8. Acknowledgements

A special thanks to Nick O'Neill for helping me get started and sending me the data I worked with.

This material is based upon work supported by the National Science Foundation under Grant No. CNS-1659144.

This work was supported by Department of Defense Grant W81XWH-13-1-077 and the National Institutes of Health Grant NS059622 to J.L.B. and V.P.L.

9. References

[1] Dhrif, H., Kubat, M., and Wuchty, S. COMB-PSO for selecting the smallest subsets of genes with highest classification performance. *Bioinformatics*

[2] Al-Ali, H., et al. (2015). Rational Polypharmacology: Systematically Identifying and Engaging Multiple Drug Targets to Promote Axon Growth. *ACS Chemical Biology*, 10, 1939-1951