Binary Simulation Optimization for Feature Selection

Ethan Houser & Dr. Sara Shashaani

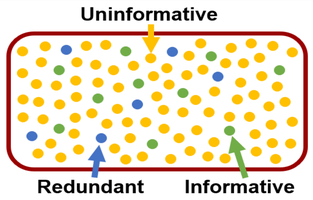
NC STATE
EDWARD P. FITTS DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING

1. Feature Selection, an NP-Hard Problem

Feature Selection (FS) is the process of **eliminating irrelevant or redundant variables** in a dataset to construct ***interpretable prediction models***. It can also be used for dimension reduction in optimization.

Solving FS with optimization is often done **greedily** and **inefficiently**.

The total number of possible variable combinations **grows exponentially** as a dataset grows in size, creating new challenges for existing methods of FS.

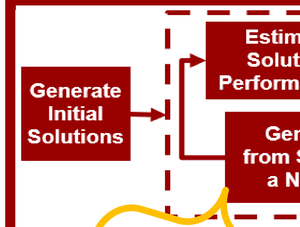


Total Features	Total Solutions
15	32768
150	1.43×10^{45}
250	1.81×10^{75}

2. FS as a Simulation Optimization

- $\min_{\theta \in \{0,1\}^d} f(\theta) := E[L(\theta)]$ — **Minimize the expected loss (prediction error) of predicting the response via features in $\boldsymbol{\theta}$.**
- $\theta_{k,i}$ — i^{th} solution sampled in iteration k.
- m_k — Total solutions sampled in iteration k.
- $L_j(\theta_{k,n})$ — j^{th} realized prediction error of the i^{th} solution $\forall j = 1, \dots, N_{k,j}$
- $\bar{L}(\theta_{k,i}) = \frac{1}{N_{k,i}} \sum_{j=1}^{N_{k,i}} L_j(\theta_{k,n})$ — **Sample average approximation** with random iid mutually exclusive training and test sets.
- $\hat{f}_k = \min_{i \in 1,\dots,m_k} \bar{L}(\theta_{k,i})$. (*) — Best estimated performance among all sampled solutions at the end of iteration k.

3. Rapid Dimension Reduction with Screening



Rank	Times Selected
1. x_1	10
2. x_7	9
3. x_4	8
4. x_2	7
5. x_5	7
6. x_8	5
7. x_3	2
8. x_6	1

(Tsai, 2013)

Shrinking Neighborhood — Explore

Nearest Neighborhood — Exploit

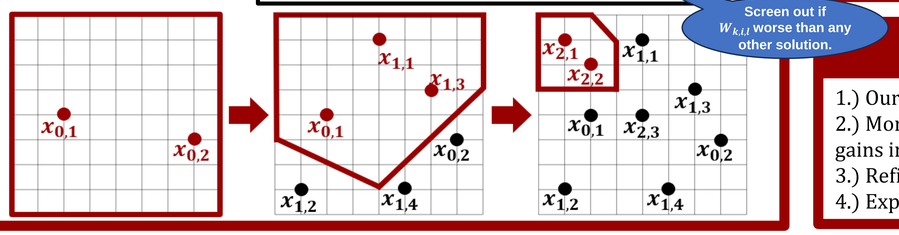
Hybrid Approach

Our Approach

1. Generate m_k new solutions and set $V_k = V_{k-1} \cap \{\theta_{k,1}, \theta_{k,2}, \dots, \theta_{k,m_k}\}$ $\quad N_{k,i} = \lceil n_o \gamma^k \rceil$
2. Compute $\bar{L}(\theta_{t,i}) \forall i = 1,2,\dots,m_k$ and $\forall t \leq k$ with $N_{k,i}$ runs
3. Compute the cushion: $W_{k,i,l} = \sqrt{\frac{t_i^2 S_{k,i}^2}{N_{k,i}} + \frac{t_l^2 S_{k,l}^2}{N_{k,l}}}$ $\forall i, l$ *and* $i \neq l$

   where $t_i = t_{(1-\alpha)^{1/m_k}, N_{k,i}-1}$ and $S_{k,i}^2 = \left(\frac{\sum_{j=1}^{N_{k,i}} \left(L_j(\theta_{k,n}) - \bar{L}(\theta_{k,i})\right)^2}{N_{k,i}} \right)$
4. Remove inferior solutions: $E_k = \{\theta_{t,i} \in V_k : \exists \theta_{t,l} \in V_k, \bar{L}(\theta_{t,i}) - \bar{L}(\theta_{t,l}) \leq -W_{k,i,l} \; \forall t, i, l\}$. Set $V_{k+1} = V_k \backslash E_k$.

Screen out if $W_{k,i,l}$ worse than any other solution.



4. Most Promising Region (MPR) in the Reduced Feature Space (with Ranked Features): Nested Partitioning

Adaptation of Algorithm by (Ólafsson, 2004) which was not efficient in higher dimensions.

Initialize: $k = 0, \Theta_0 = \{0,1\}^d, \{x_{[1]}, x_{[2]}, \cdots, x_{[p]}\}$ surviving ranked features from screening with $p \ll d$

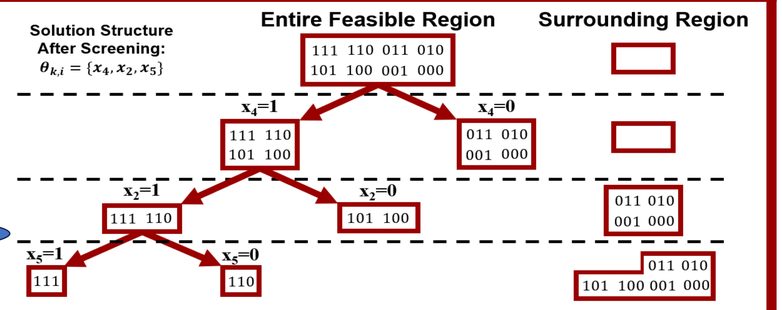
For $k = 1,2,\cdots$

1. **Partitioning**: Set subregions Θ_k^1, Θ_k^2, and Θ_k^3 such that
   - $\Theta_k^1 \cup \Theta_k^2 = \Theta_k$ (the current MPR) where $x_{[k]} = 1$ in Θ_k^1 and $x_{[k]} = 0$ in Θ_k^2,
   - $\Theta_k^3 = \Theta \setminus \Theta_{k-1}$ (i.e., $\Theta_k^1 \cap \Theta_k^2 \cap \Theta_k^3 = \emptyset$).

   Partition using the k-th best feature in the ranked list from screening.
2. **Sampling**: Draw m_k^r samples uniformly from Θ_k^r, and compute $\hat{f}_k^r$ similar to (*) for $r = 1,2,3$.
   - with $N_k^r = \left\lceil \frac{h^2 (S_{k,j}^r)^2}{\delta^2} \right\rceil$ runs (Ólafsson, 2004)

   If the best performance was observed in the new partitions rather than the surrounding region, partition them further
3. **Update MPR**: If $\text{argmin}_{r \in \{1,2,3\}} \{\hat{f}_k^r\} < 3$, then let $\Theta_{k+1} = \Theta_k^r$ be the next MPR.

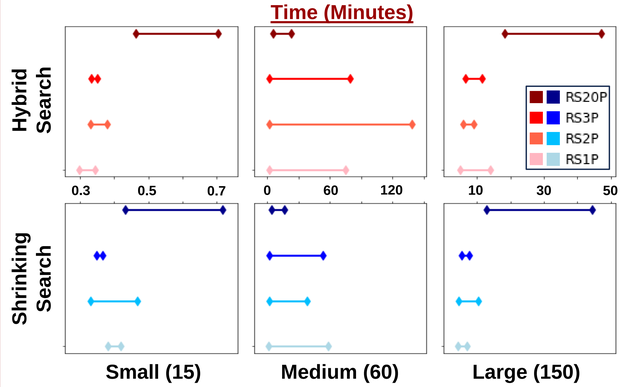
Otherwise, implement a **backtracking** strategy to return to a parent node of Θ_{k-1}.



5. Experimentation with Synthetic Datasets of Varying Sizes with Many Noisy and Redundant Features

5.1. Effect of the Number of Screening Steps

Time (Minutes)

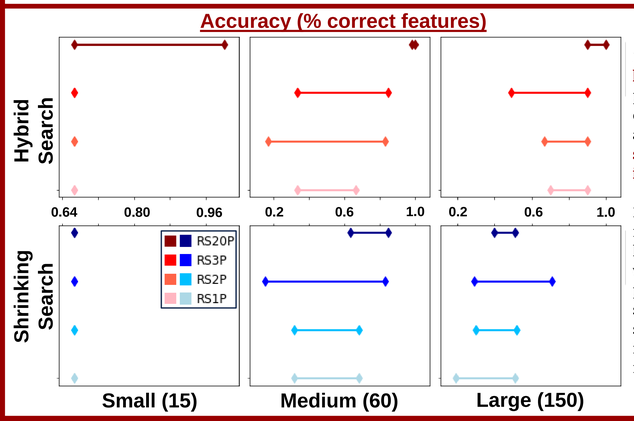


Summary

1.) Hybrid and shirking neighborhood search exhibit similar run time.

2.) In a few screening steps, there is not much of a difference in time, but as more screening steps are taken, the algorithm's run time increases.

Accuracy (% correct features)



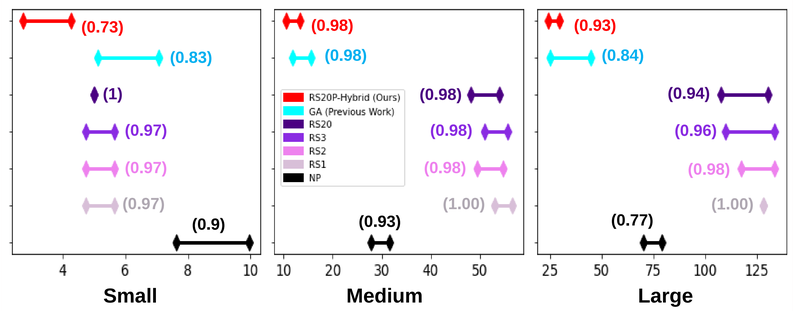
Summary

1.) Dominance in the **hybrid search** procedure as the size of the **dataset grows** and as the number of **screening steps increases**.

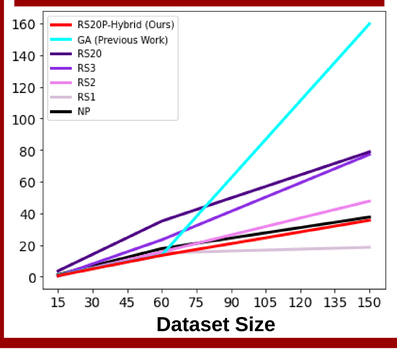
2.) The top performing model for medium and large dataset is one with hybrid search procedure over 20 screening steps. For small dataset, only a few screening steps are more reliable.

5.2. Comparison with State of the Art

80% Prediction Interval of Total Features (Accuracy %)



Time (Minutes)



As Data Size Increases:

	Time	Accuracy	Total Features
RSP (Ours)	✓	✓	✓
GA (Previous Work)	✗	✓	✓
RS (Low # Steps)	✓	✓	✗
RS (High # Steps)	⚠	✓	✗
NP	✓	✗	⚠

6. Concluding Remarks

1.) Our approach outperforms existing methods especially in larger datasets.
2.) More **exploitation** in screening leads to a marginal increase in **time** but significant gains in **accuracy**.
3.) Refinement to **reduce computational effort** while maintaining performance.
4.) Experimentation with increasingly **larger and real datasets.**

7. References

1.) Tsai, S. C. Rapid screening procedures for zero-one optimization via simulation. *Informs Journal on Computing*, *25*(2), 317-331. (2013)

2.) Ólafsson, S. Two-Stage Nested Partitions Method for Stochastic Optimization. *Methodology and Computing in Applied Probability* **6**, 5–27 (2004)