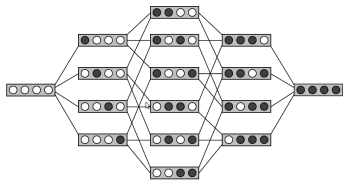


Introduction to Machine Learning

Feature Selection

Feature Selection: Wrapper methods



Learning goals

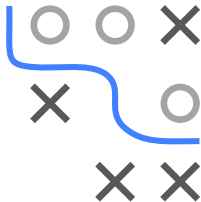
- Understand how wrapper methods work
- Forward + backward search, EAs
- Advantages and disadvantages

OBJECTIVE FUNCTION

Given p features, **best-subset selection problem** is to find subset $S \subseteq \{1, \dots, p\}$ optimizing objective $\Psi : \Omega \rightarrow \mathbb{R}$:

$$S^* \in \arg \min_{S \in \Omega} \{\Psi(S)\}$$

- Ω = search space of all feature subsets $S \subseteq \{1, \dots, p\}$. Usually we encode this by bit vectors, i.e., $\Omega = \{0, 1\}^p$ (1 = feat. selected)
- Objective Ψ can be different functions, e.g., AIC/BIC for LM or cross-validated performance of a learner
- Poses a discrete combinatorial optimization problem over search space of size $= 2^p$, i.e., grows exponentially in p (power set)
- Unfortunately can not be solved efficiently in general (NP hard; see, e.g., [Natarajan 1995](#))
- Can avoid searching entire space by employing efficient search strategies, traversing search space in a “smart” way

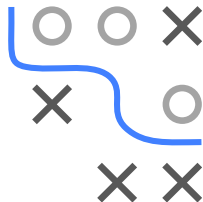
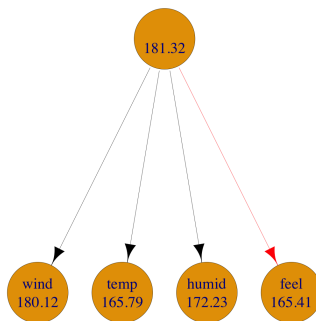


GREEDY FORWARD SEARCH

Let $S \subset \{1, \dots, p\}$ be subset of feature indices.

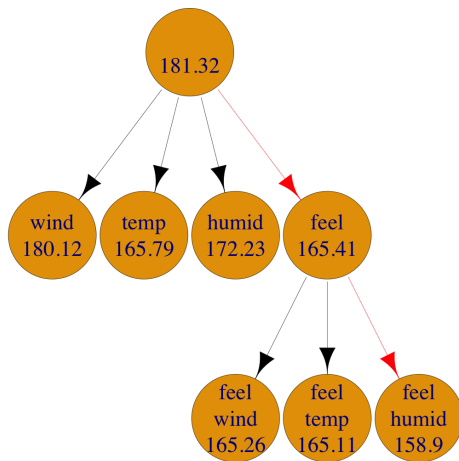
- 1 Start with the empty feature set $S = \emptyset$
- 2 For a given set S , generate all $S_j = S \cup \{j\}$ with $j \notin S$.
- 3 Evaluate the classifier on all S_j and use the best S_j

Example GFS on a subset of bike sharing data with features windspeed, temp., humidity and feeling temp. Node value is RMSE.



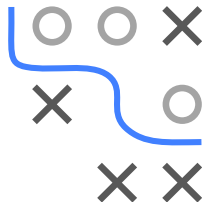
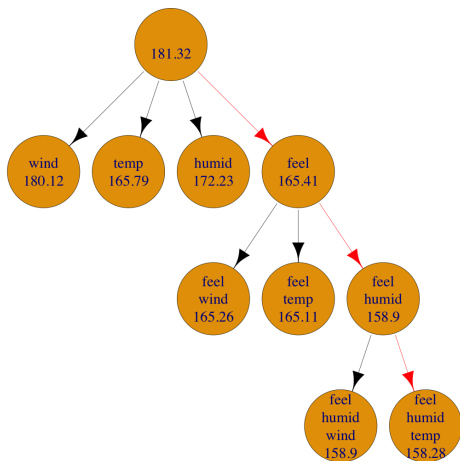
VISUALIZATION OF GFS

- 4 Iterate over this procedure

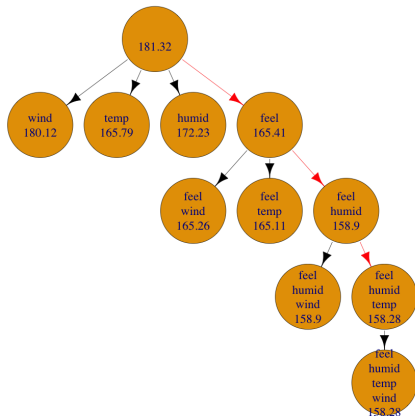


VISUALIZATION OF GFS

- ④ Iterate over this procedure



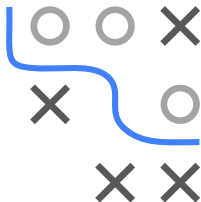
VISUALIZATION OF GFS



- 5 Terminate if performance does not improve further or max. number of features is used

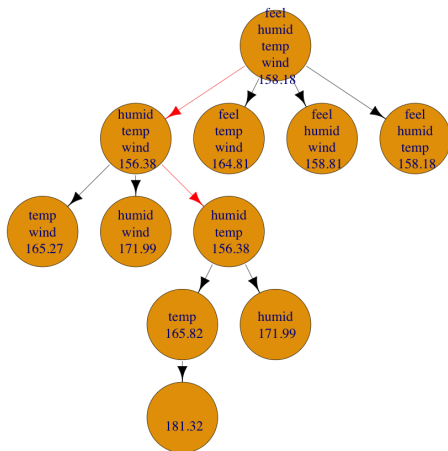
GREEDY BACKWARD SEARCH

- Start with the full index set of features $S = \{1, \dots, p\}$.
- For a given set S generate all $S_j = S \setminus \{j\}$ with $j \in S$.
- Evaluate the classifier on all S_j and use the best S_j .
- Iterate over this procedure.
- Terminate if:
 - the performance drops drastically, or
 - falls below given threshold.
- GFS is much faster and generates sparser feature selections
- GBS much more costly and slower, but sometimes slightly better.



VISUALIZATION OF GBS

Example Greedy Backward Search on bike sharing data



EXTENSIONS

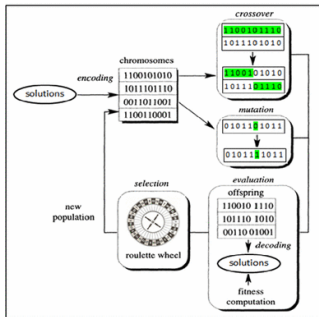
- Eliminate or add multiple features at once to increase speed
- Allow alternating forward and backward search (also known as stepwise model selection by AIC/BIC in statistics)
- Randomly sample candidate feature subsets in each iteration
- Focus search on regions of feature subsets where an improvement is more likely



EXTENSIONS: GENETIC ALGORITHMS FOR FS

Example Template for $(\mu + \lambda)$ -Evolutionary Strategy applied to FS

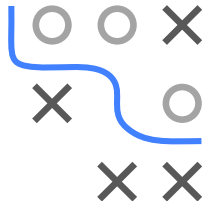
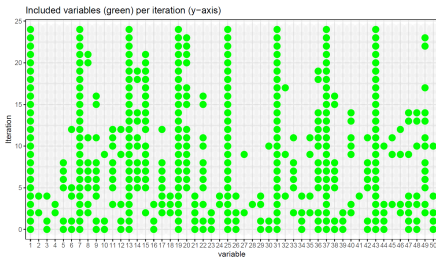
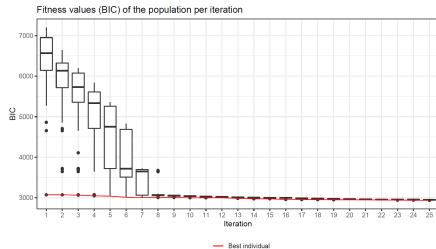
- 1 Initialization: μ random bit vectors (feature inclusion/exclusion)
- 2 Evaluate model performance for bit vectors
- 3 Select μ fittest bit vectors (parents)
- 4 Generate λ offspring applying crossover and mutation
- 5 Select μ fittest bit vectors from $(\mu + \lambda)$ options for next generation
- 6 Repeat steps 2-5 until stopping criterion is met



- Use CV/validation set for evaluation to avoid overfitting
- Choice of μ and λ allows some control over exploration vs. exploitation trade-off
- See our [slds-lmu 2021](#) for further information



EXTENSIONS: GENETIC ALGORITHMS FOR FS



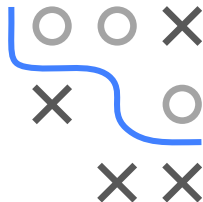
Top: BIC over number of iterations.

Bottom: Bit representation of selected features over iterations.

WRAPPERS

Advantages:

- Can be combined with any learner
- Any performance measure can be used
- Optimizes the desired criterion directly



Disadvantages:

- Evaluating target function is expensive
- Does not scale well with number of features
- Does not use additional info about model structure
- Nested resampling becomes necessary