Correlation Based Feature Selection (CFS)

- Looks at feature subsets to find uncorrelated features
- Creates subsets with best-first search, starting with one feature.
- Stops after 5 expansions lead to no improvement.
Goodness of feature subset

\[ Merit_s = \frac{k\bar{r}_{cf}}{\sqrt{k + k(k - 1)\bar{r}_{ff}}} \]

Where \( s \) is the feature subset with \( k \) features, \( \bar{r}_{cf} \) is the average feature class correlation and \( \bar{r}_{ff} \) is the average feature-feature intercorrelation.
Continuous/numeric features are discretized using Fayyad’s method.

Symmetric uncertainty is used to measure the degree of association between features

\[ SU = 2 \times \frac{H(X) + H(Y) - H(X,Y)}{H(X) + H(Y)} \]

Where \( H(X) \) is the marginal entropy and \( H(X,Y) \) is the joint entropy of \( X \) and \( Y \).
Locally predictive feature heuristic

After feature selection stops: A feature gets into the subset if its correlation to the class is higher than the highest correlation between it and any already selected feature.
In the paper describing the algorithm, it says that a discretized copy of each training split was made and given to the learning algorithm!
CFS issues

- The discretization of the attributes is quite critical. Done poorly and poor results.

- Also, one should use the given discretization in training/testing to really get the most out of the feature selection!

- If you use the features without discretization, will it help as much?
RELIEFF

A filter algorithm for Feature Selection
What it does

- Independently of any classifier. Select the $k$ nearest neighbors to a randomly chosen example, *from each class*.

- We would like attributes to have similar values for the nearest neighbors in the class.

- We would like attributes to have different values for nearest neighbors from other classes (differentiate well).
Goal: Rank attributes/features

- We need a formula that penalizes attributes with different values for the close examples from the same class and rewards different values from a different class.
**Algorithm ReliefF**

- **Input:** for each training instance a vector of attribute values and the class value
- **Output:** the vector $W$ of estimations of the qualities of attributes

1. set all weights $W[A] := 0.0$;
2. for $i := 1$ to $m$ do begin  // for $m$ randomly chosen examples
3. randomly select an instance $R_i$;
4. find $k$ nearest hits $H_j$;  // from same class
5. for each class $C .ne. class(R_i)$ do
6. from class $C$ find $k$ nearest misses $M_{j}(C)$;  // $1 \leq j < |C|$  
7. for $A := 1$ to $a$ do  // $a$ is the number of attributes
8. $W[A] := W[A] - \frac{\sum_{j=1}^{k} \text{diff}(A, R_i, H_j)}{m \cdot k} + \sum_{C \neq \text{class}(R_i)} \left[ \frac{P(C)}{1 - P(\text{class}(R_i))} \sum_{j=1}^{k} \text{diff}(A, R_i, M_{j}(C)) \right] / (m \cdot k)$

9. end;

$$W[A] := W[A] - \frac{\sum_{j=1}^{k} \text{diff}(A, R_i, H_j)}{m \cdot k} + \sum_{C \neq \text{class}(R_i)} \left[ \frac{P(C)}{1 - P(\text{class}(R_i))} \sum_{j=1}^{k} \text{diff}(A, R_i, M_{j}(C)) \right] / (m \cdot k)$$
For nominal attributes, where $I_1$ and $I_2$ are instances.

$$\text{diff}(A, I_1, I_2) = \begin{cases} 0 & : \text{value}(A, I_1) = \text{value}(A, I_2) \\ 1 & : \text{Otherwise} \end{cases}$$

For continuous attributes

$$\text{diff}(A, I_1, I_2) = \frac{|\text{value}(A, I_1) - \text{value}(A, I_2)|}{\text{max}(A) - \text{min}(A)}$$

For nearest neighbors we use the sum of the attribute distances.
=== Run information ===

Evaluator: weka.attributeSelection.ReliefFAttributeEval -W -M 100 -D 44 -K 10 -A 2
Search:weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1
Relation: iris
Instances: 150
Attributes: 5
  - sepal length
  - sepal width
  - petal length
  - petal width
  - class
Evaluation mode: evaluate on all training data

=== Attribute Selection on all input data ===

Search Method: Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 5 class):
  ReliefF Ranking Filter, Instances sampled: 100, Number of nearest neighbours (k): 10
  Exponentially decreasing (with distance) influence for nearest neighbours. Sigma: 2

Ranked attributes: 0.37 4 petal width, 0.342 3 petal length, 0.141 2 sepal width, 0.135 1 sepal length

Selected attributes: 4,3,2,1 : 4