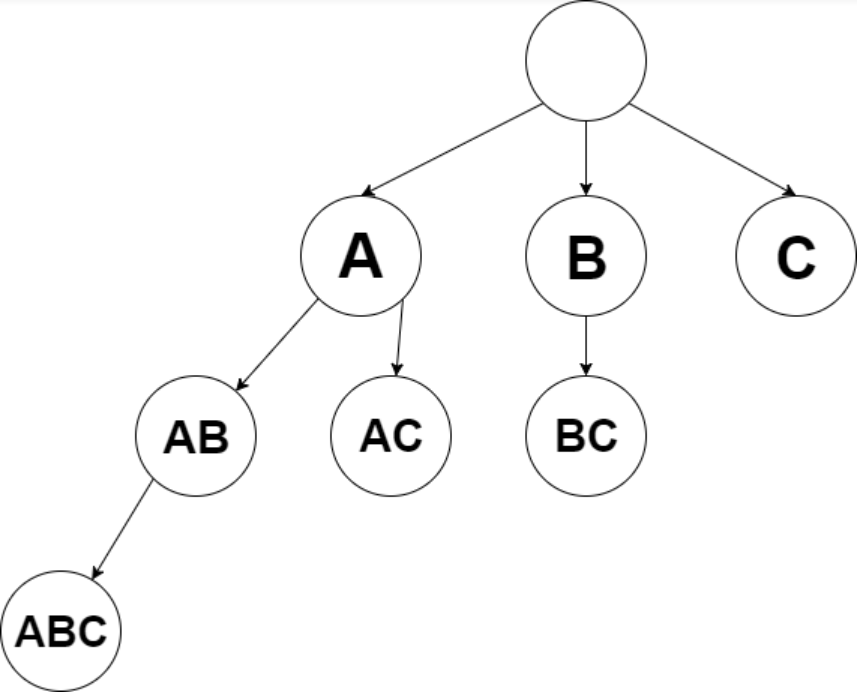
**Group13: Feature Selection using graph search**

Yiran Tian (20580120), YuCheng Liu(20558944) , Tian zhu(20517833), Conghao He(20589010)

**a) Justifying the choice of the algorithm(s)**

The algorithm we choose is Breadth First Search. For the feature selection problem, a tree is generated with each nodes representing a subset of all the features available. A cost function (regression) has been defined to evaluate the correlation of feature clusters at each node. Then, each node may have to be traversed before finding the feature subset giving highest score. Therefore, an uninformed graph search algorithm is preferred as the best node’s location is unknown and no heuristic information is available. Since cost function is only evaluated at each node and a goal node that reaches the targeted value for the cost function may appear before the the maximum depth, BFS is a proper choice of algorithm as it is complete and find the shallowest goal state.

**b) Discussing algorithm complexity as it relates to the problem at hand** 

According to the tree structure built in this problem, no cycles are present and no child node has more than one parent. Even though BFS returns the goal node, or the one that maximizes the cost function may still be the last node to be traversed. Therefore, the worst case time complexity is O(V), where V denotes number of nodes. Similarly, the worst case space complexity is also O(V). Ttree nodes correspond to different subset of features. The number of total combinations (V) is 2^N, where N represents total number of features. Notice the root of the tree is empty.

**c) Discuss what will happen if you increase your search space**

When the search space increases, where the clusters of features increases for feature selection specifically, it will take longer to access all the nodes. The increase will be exponential since the space and time complexity is order of V, where V = 2^N. When the search space is expanded, the chance that a better feature subset that maximizes the cost function may be increased. If no such node (subset) is found, solution will not be affected since BFS was used for node traversal and not for path finding.

**d) Discuss the solution(s) reached by the algorithm based on your dataset and its optimality**

The highest regression score achieved by the features sets is 97. There are 36 sets of features obtain the same score. The data set selected was “Breast Cancer Wisconsin”, which has 30 features in total. By statistics, only 3.35\*10^-6 % of possible feature combinations are selected. BFS covers all nodes, and provides thorough check. However, as the BFS is only used for node traversal; thus, the complexity for tree traversal is always O(V).The optimality of BFS is not well shown according to the feature selection problem. There were three features that were most dominant. The initial score with all 30 features is 67. The score achieved by adding additional features on these three features did not increase/decrease the accuracy, thus proving these three features having strong revelevancy while the additional features added are identified as non-relevant features. For instance, 3 best outcomes are listed below. It could be seen that the first solution is a subset of following solutions, therefore the 5 features are most important.

For example:

Features se t1 : ['area\_worst', 'compactness\_worst', 'perimeter\_worst', 'radius\_worst', 'texture\_worst'],

Features set 2 : ['area\_worst', 'compactness\_worst', 'concavity\_se', 'perimeter\_worst', 'radius\_worst', 'texture\_worst']

Features set 3 ['area\_worst', 'compactness\_worst', 'concave points\_worst', 'concavity\_worst', 'fractal\_dimension\_se', 'perimeter\_worst', 'radius\_worst', 'smoothness\_worst', 'texture\_worst']

All of those features set have a score of 97.2% accuracy.