**Programme** : B. Sc.
**Subject** : Computer Science
**Semester** : VI
**Course Code** : CSD107
**Course Title** : Introduction to Data Analytics

**Title of the Unit** : Feature Generation and Feature Selection
**Module Name** : Feature Selection Algorithms part 2
**Module Number** : 14
**Name of the Presenter** : Ms. Sneha Dessai, Assistant Professor in Computer Science, GCQ-Goa

## Feature selection algorithms:

There are three primary methods of stepwise regression:

- forward selection
- backward elimination
- combined approach – bidirectional search

### 1. Forward selection

In forward selection you start with a regression model with no features, and gradually add one feature at a time according to which feature improves the model the most based on a selection criterion. This looks like this: build all possible regression models with a single predictor. Pick the best.

Pick the best of those. You keep adding one feature at a time, and you stop when your selection criterion no longer improves, but instead gets worse.

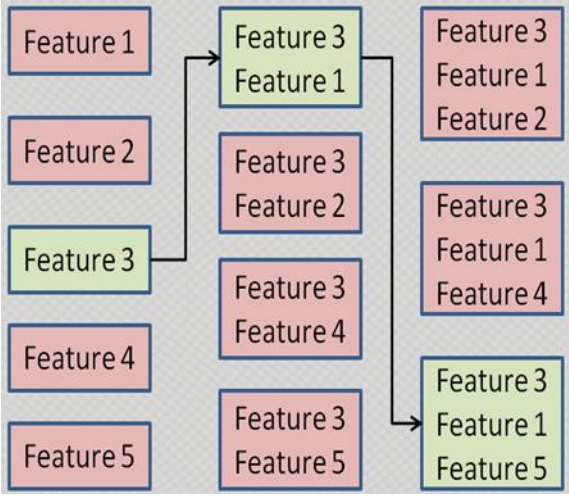
Sequential forward selection :

First, the best single feature is selected (i.e., using some objective function). Then, pairs of features are formed using one of the remaining features and this best feature, and the best pair is selected.

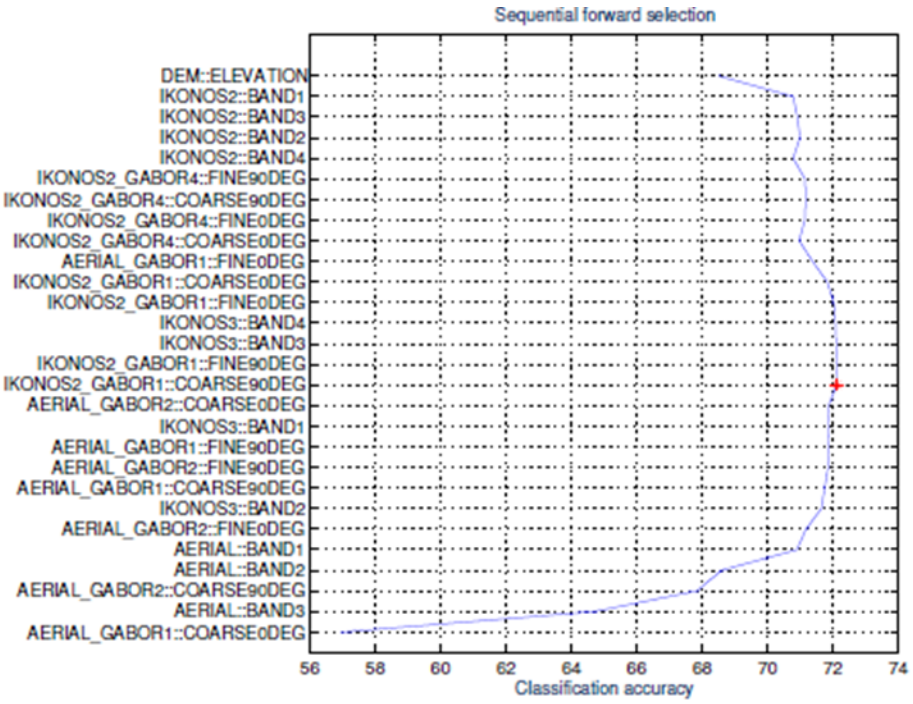
Next, triplets of features are formed using one of the remaining features and these two best features, and the best triplet is selected. This procedure continues until a predefined number of features are selected.

Algorithm : Sequential Forward feature set generation – SFG

```
function  SFG ( F – full set, U – measure)
    initialize S = {}                    >> S stores the selected features
    repeat
            f = FINDNEXT(F)
            S = S U {f}
            F = F – {f}
    until S satisfies U or F = {}
    return S
end function
```




Results of sequential forward feature selection for classification of a satellite image using 28 features. x-axis shows the classification accuracy (%) and y-axis shows the features added at each iteration (the first iteration is at the bottom). The highest accuracy value is shown with a star.




**2. Backward elimination**

In backward elimination you start with a regression model that includes all the features, and you gradually remove one feature at a time according to the feature whose removal makes the biggest improvement in the selection criterion. You stop removing features when removing the feature makes the selection criterion get worse.

Sequential backward elimination

First, the objective function is computed for all n features. Then, each feature is deleted one at a time, the objective function is computed for all subsets with n-1 features, and the worst feature is discarded.

Next, each feature among the remaining n-1 is deleted one at a time, and the worst feature is discarded to form a subset with n-2 features. This procedure continues until a predefined number of features are left.

Algorithm : Sequential backward feature set generation – SBG

function SFG ( F – full set, U – measure)

initialize S = {} > S holds the removed features

repeat

f = GETNEXT(F)

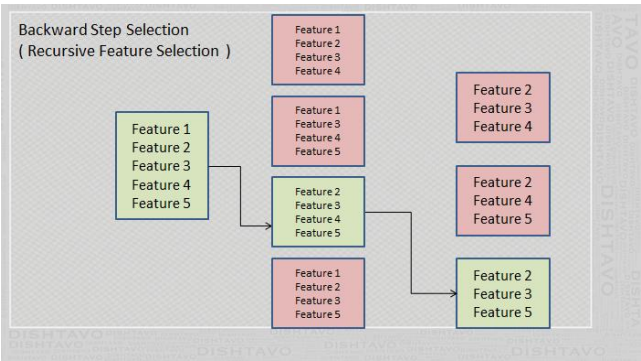
F = F – {f}

S = S U {f}

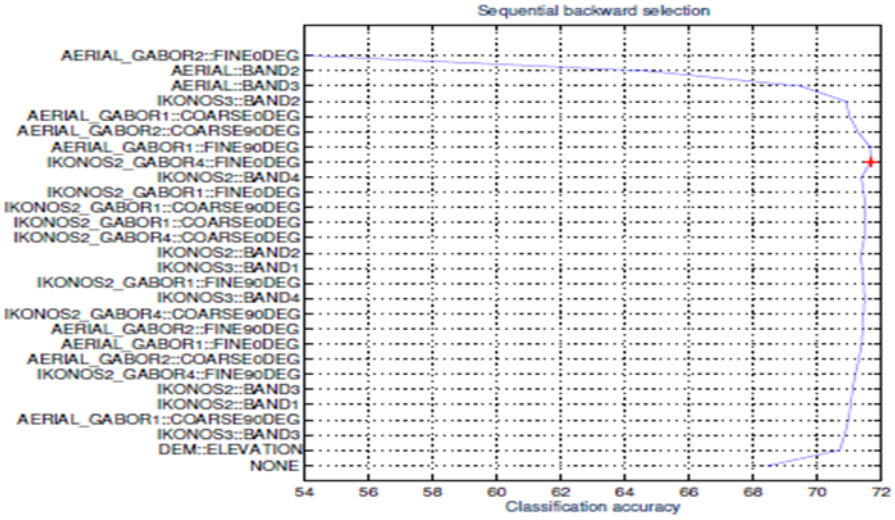
until S does not satisfy U or F = {}

return F U {f}

end function




Results of sequential backward feature selection for classification of a satellite image using 28 features.



3. Combined approach - Bidirectional Search
Bidirectional search applies Sequential Forward Selection and Sequential Backward Selection simultaneously.

Sequential Forward Selection is performed from the empty set.

Sequential Backward Selection is performed from the full set.

To guarantee that Forward and Backward Selection converge to the same solution:

Features already selected by the Forward search are not removed by Backward search.

Features already removed by Backward search are not added by Forward search.

Limitations of Sequential Forward Selection and Sequential Backward Selection :

The main limitation of Sequential Forward Selection is that it is unable to remove features that become non useful after the addition of other features.

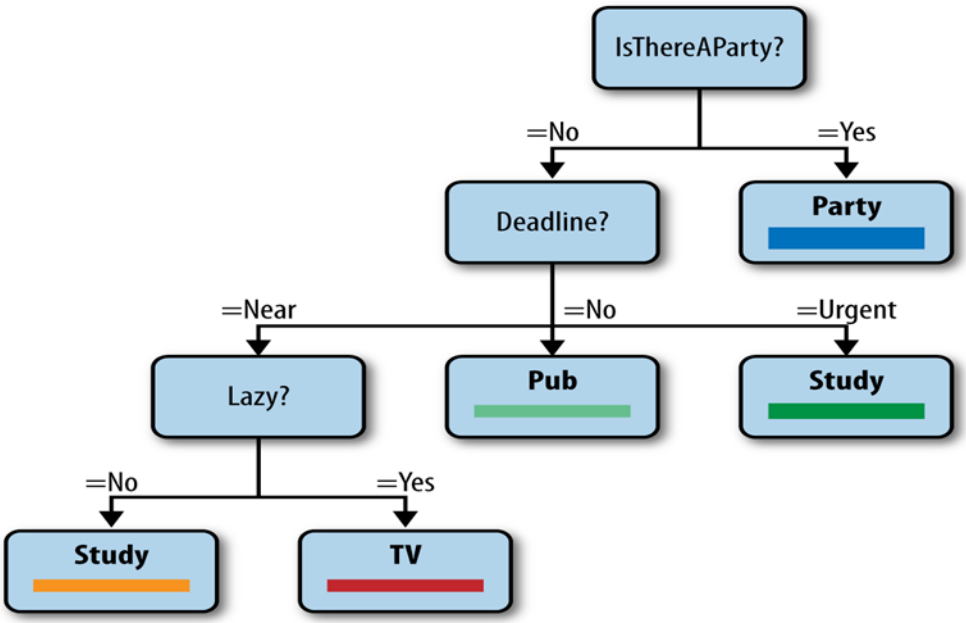
The main limitation of Sequential Backward Selection is its inability to reevaluate the usefulness of a feature after it has been discarded.

**3. Feature Selection – Embedded Methods : Decision Trees**

In embedded methods, you don’t need to use a filter here because the information gain method is doing your feature selection for you.

Decision trees have an intuitive appeal because outside the context of data science in our every day lives, we can think of breaking big decisions down into a series of questions.

See the decision tree in Figure about a college student facing the very important decision of how to spend their time.




This decision is actually dependent on a bunch of factors:

Whether or not there are any parties or deadlines.

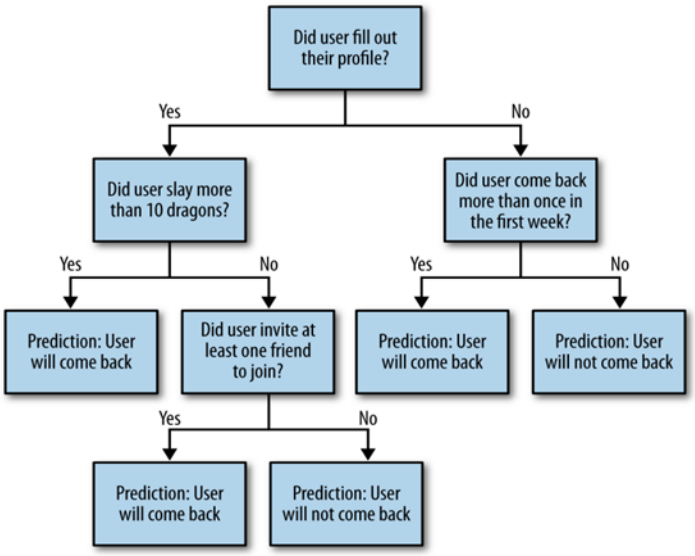
How lazy the student is feeling.

What they care about most (parties).

For the Chasing Dragons example, you want to classify users as “Yes, going to come back next month” or “No, not going to come back next month.”

The class of any given user is dependent on many factors (number of dragons the user slew, their age, how many hours they already played the game).

Choosing a feature which is the most informative first




Decision trees

A decision tree takes as input an object or situation described by a set of attributes and returns a decision.

This decision is the predicted output value for the input.

A decision tree reaches its decision by performing a sequence of tests.

All non leaf nodes lead to partial decisions and assist in moving towards the leaf node.

Leaf nodes depict the decision about a character having attributes falling on the path from the root node

Each example that participate in the construction of the decision tree is called a training data and the complete set of the training data is called as training set.

Decision Tree Algorithm:

You build your decision tree iteratively, starting at the root.

You need an algorithm to decide which attribute to split on, e.g. which node should be the next one to identify.

You choose that attribute in order to maximize information gain.

You keep going until all the points at the end are in the same class or you end up with no features left. In this case, you take the majority vote.