Data Pre-Processing-III

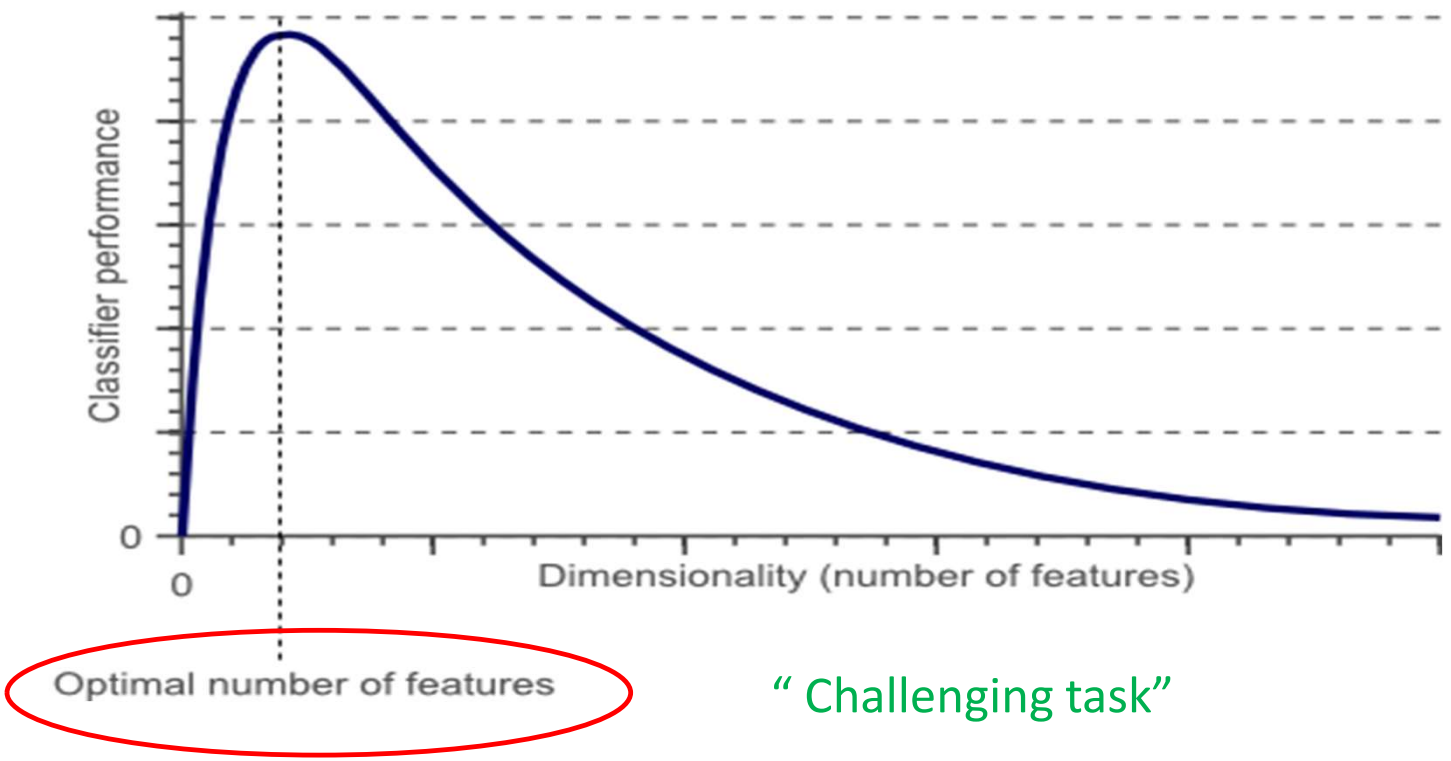
(Data Reduction: Introduction, Feature Selection)

Dr. JASMEET SINGH
ASSISTANT PROFESSOR, CSED
TIET, PATIALA

Dimensionality/Data Reduction

- The number of input variables or features for a dataset is referred to as its dimensionality.
- **Dimensionality reduction** refers to techniques that reduce the number of input variables in a dataset.
- More input features often make a predictive modeling task more challenging to model, more generally referred to as the ***curse of dimensionality***.
- There exist a optimal number of feature in a feature set for corresponding Machine Learning task.
- Adding additional features than optimal ones (strictly necessary) results in a performance degradation (because of added noise).

Dimensionality/Data Reduction



" Challenging task"

Dimensionality/Data Reduction

Benefits of data reduction

- Accuracy improvements.
- Over-fitting risk reduction.
- Speed up in training.
- Improved Data Visualization.
- Increase in explain ability of ML model.
- Increase storage efficiency.
- Reduced storage cost.

Data Reduction Techniques

Feature Selection –

find the best set of feature

- Filter methods
- Wrapper methods
- Embedded methods

Feature Extraction-

methods of constructing combinations of the variables to get around these problems while still describing the data.

- Principal Component Analysis
- Singular-Valued Decomposition
- Linear Discriminant Analysis

Feature Selection

- Feature selection in machine learning is to find the best set of features that allows one to build useful models of studied phenomena.
- The two key drivers used in feature selection are:
  - **Maximizing feature relevance**
    - Feature contributing significant information for the machine learning model – strongly relevant
    - Feature contributing little information for the machine learning model – weakly relevant
    - Feature contributing no information for the machine learning model – irrelevant
  - **Minimizing feature redundancy**
    - Information contributed by the feature is similar to the information contributed by one or more other features.

Feature Selection (Contd….)

Roll Number	Age	Height	Weight

- Let us consider a student database, with attributes Roll Number, Age , Height and Target Variable (Weight). The objective is to predict a weight for each new test case.
- Roll Number is irrelevant as it will not provide any information regarding weight of students.
- Age and Height are redundant as both provide same information.

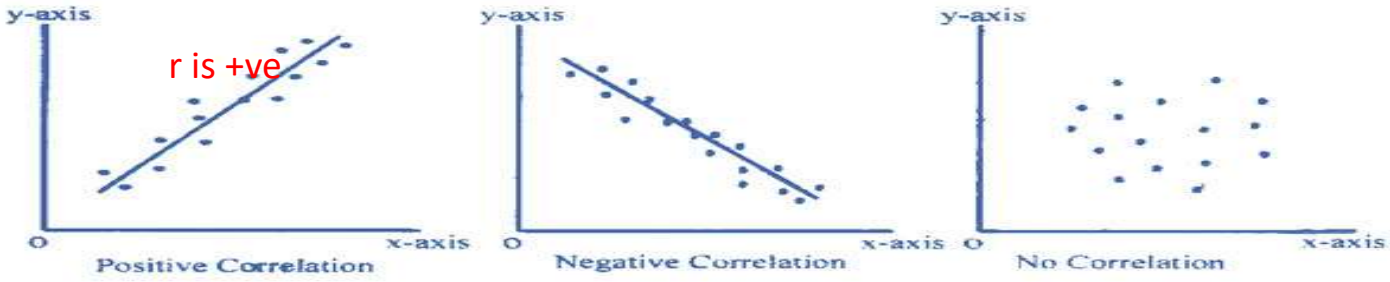
Feature Selection- Measuring Feature Redundancy

- Feature Redundancy is measured in terms of similarity information contributed by features.
- Similarity information is measured in terms of:
  - Correlation-based features.
  - Distance based features.

Feature Selection- Measuring Feature Redundancy

- To deal with redundant features correlation analysis is performed. Denoted by r.
- A threshold is decided to find redundant features.

$$r = \frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\sqrt{[n\Sigma x^2 - (\Sigma x)^2][n\Sigma y^2 - (\Sigma y)^2]}}$$



Feature Selection- Measuring Feature Redundancy

Distance-based:

- The most commonly used distance metric is various forms of **Minkowski distance.**

$$d(F_1, F_2) = \sqrt[r]{\sum_{i=1}^{n}(F_{1i} - F_{2i})^r}$$

It takes the form of **Euclidian distance** when r =2 (L_2 norm) and **Manhattan distance** when r = 1 (L_1 norm).

- **Cosine similarity** is another important metric for computing similarity between features.

$$cos(F_1, F_2) = \frac{F_1 \,.\, F_2}{|F_1\,||F_2|}$$

Where F_1 and F_2 denote feature vectors.

Feature Selection- Measuring Feature Redundancy

For binary features, following metrics are useful:

1. Hamming distance: number of values which are different in two feature vectors.
2. Jaccard distance: 1- Jaccard Similarity

$$Jaccard\ Similarity = \frac{n_{11}}{n_{01} + n_{10} + n_{11}}$$

3. Simple Matching Coefficient (SMC):

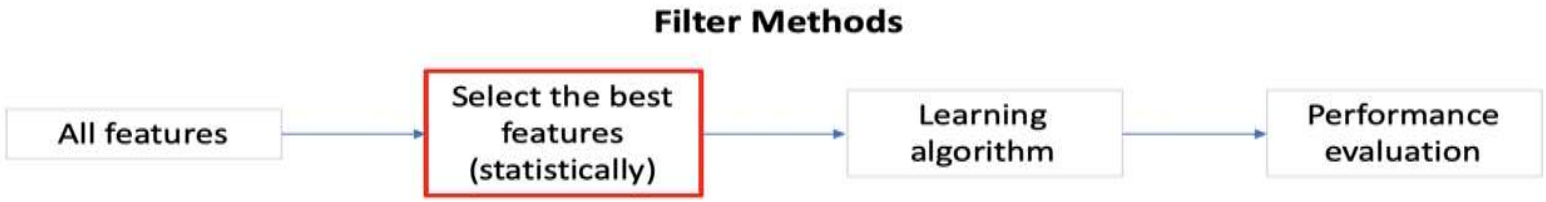
$$SMC = \frac{n_{11} + n_{00}}{n_{00} + n_{01} + n_{10} + n_{11}}$$

Where n_{11}, n_{00} represent number of cases where both features have value 1 and 0 respectively
n_{10} denote cases where feature 1 has value 1 and feature 2 has value 0.
n_{01} denote cases where feature 1 has value 0 and feature 2 has value 1.

Feature Selection Approaches

Filter Approach:

- In this approach, the feature subset is selected based on statistical measures.
- No learning algorithm is employed to evaluate the goodness of the feature selected.
- Commonly used metrics include correlation, chi square, Fisher score, ANOVA, Information Gain, etc.



Chi-Square Test for Feature Selection

- A chi-square test is used in statistics to test the independence of two events.
- Given the data of two variables, we can get observed count O and expected count E.
- Chi-Square measures how expected count E and observed count O deviates each other.

$$\chi_c^2 = \sum \frac{(O - E)^2}{E}$$

where c is the degree of freedom, O is the observed value and E is the expected value.

- When two features are independent, the observed count is close to the expected count, thus we will have smaller Chi-Square value.
- Higher the Chi-Square value the feature is more dependent on the response and it can be selected for model training.

Chi-Square Test for Feature Selection (Contd….)

- Steps to perform the Chi-Square Test:

1. Define Hypothesis.
2. Build a Contingency table.
3. Find the expected values.
4. Calculate the Chi-Square statistic.
5. Accept or Reject the Null Hypothesis.

Chi-Square Test for Feature Selection (Contd….)

- Consider a data-set where we have to determine why customers are leaving the bank, let's perform a Chi-Square test for two variables.
- ***Gender*** of a customer with values as Male/Female as the predictor and ***Exited*** describes whether a customer is leaving the bank with values Yes/No as the response.
- In this test we will check *is there any relationship between Gender and Exited*.
- **Step 1: Define Hypothesis**

Null Hypothesis (H0): Two variables are independent.

Alternate Hypothesis (H1): Two variables are not independent.

Chi-Square Test for Feature Selection (Contd….)

- **Step 2: Contingency Table**

Exited\ Gender	Yes	No	Total
Male	38	178	216
Female	44	140	184
Total	82	318	400

Degrees of freedom for contingency table is given as (r-1) * (c-1) where r,c are rows and columns. Here df = (2–1) * (2–1) = 1.

Chi-Square Test for Feature Selection (Contd….)

- **Step 3. Find the Expected Value**

Based on the null hypothesis that the two variables are independent. We can say if A, B are two independent events

$$P(A \cap B) = P(A) * P(B)$$

Let's calculate the expected value for the first cell that is those who are Males and are Exited from the bank.

$E1 = n * p$

$p = p(Yes) * p(Male)$

$p = (82/400) * (216/400)$

$p = 0.1107$

now, $E1 = 400 * 0.1107 = 44$

Chi-Square Test for Feature Selection (Contd….)

- In similar, we calculate E2, E3, E4 and get the following results.

Exited\Gender	Yes	No
Male	44	172
Female	38	146

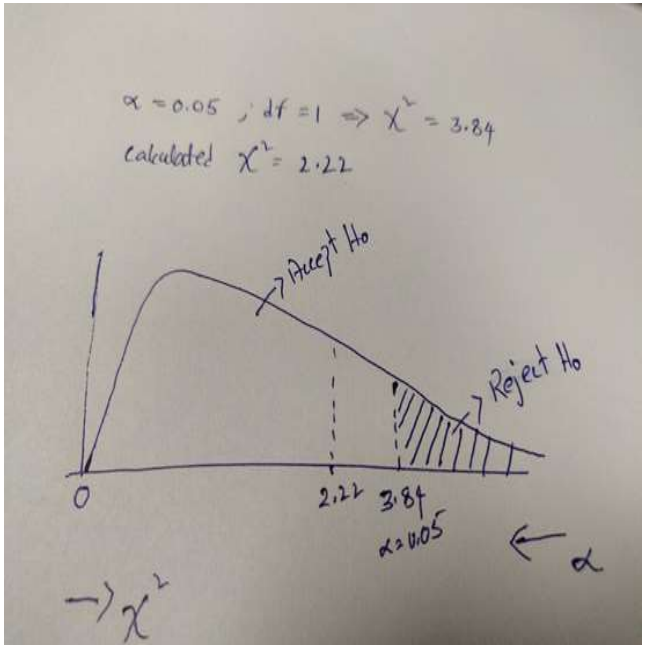
- **Step 4 Calculate Chi-Square value**: Summarizing the observed values and calculated expected values into a table and determine the Chi-Square value.

Gender,Exited	O	E	O-E	Square of O-E	(Square of O-E) / E
Male,Yes	38	44	-6	36	0.818181818
Male,No	178	172	6	36	0.209302326
Female,Yes	44	38	6	36	0.947368421
Femal,No	140	146	-6	36	0.246575342
Chi Square Value					2.221427907

Chi-Square Test for Feature Selection (Contd….)

Step 5. Accept or Reject the Null Hypothesis

- With 95% confidence that is alpha = 0.05, we will check the calculated Chi-Square value falls in the acceptance or rejection region.
- Having degrees of freedom =1(calculated with contingency table) and alpha =0.05 the Chi-Square value is 3.84.
- In the fig, we can see Chi-Square ranges from 0 to inf and alpha ranges from 0 to 1 in the opposite direction. We will reject the Null hypothesis if Chi-Square value falls in the error region (alpha from 0 to 0.05).
- So here we are accepting the null hypothesis since the Chi-Square value is less than the critical Chi-Square value.
- *To conclude the two variables are independent, Gender variable cannot be selected for training the model.*



Feature Selection using Information Gain

- The concept of Information Gain is based on:

The more we know about a topic, the less new information you are apt to get about it. To be more concise: If you know an event is very probable, it is no surprise when it happens, that is, it gives us little information that it actually happened.

- The amount of information gained is inversely proportional to the probability of an event happening.
- We can also say that as the Entropy increases the information gain decreases. This is because Entropy refers to the probability of an event.

Feature Selection using Information Gain

- Information Gain = Entropy of Parent – sum (weighted % * Entropy of Child)

Weighted % = Number of observations in particular child/sum (observations in all child nodes)

- In particular, Information Gain for a feature column **A** is calculated as:

$$Information\ Gain(S, A) = Entropy(S) - \sum_{v=1}^{|v|} \frac{|S_v|}{|S|} Entropy(S_v)$$

where $\mathbf{S_v}$ is the set of rows in **S** for which the feature column **A** has value **v**, $|\mathbf{S_v}|$ is the number of rows in $\mathbf{S_v}$ and likewise $|\mathbf{S}|$ is the number of rows in **S.**

- **Information Gain calculates the reduction in the entropy and measures how well a given feature separates or classifies the target classes.**
- **The feature with the highest Information Gain is selected as the best one.**

Feature Selection using Information Gain- Example

A labeled dataset contains three input features Gender, Car Type, and House Type about an individual. On the basis of these input features the individuals are labeled as High Income or Medium Income group. The dataset contains 20 training examples; 10 of each output class. The feature Gender has two unique values Male and Female, the feature Car Type has three unique values Sports, Luxury, Family and the feature House Type has three unique values Small, Medium, Large. The distribution of these values across High Income and Medium Income group for 20 individuals is shown in Table 5. Using Information Gain, select which two input features are important for the classification model.

Attribute: Gender			Attribute Car Type			Attribute: House Type		
	High Income	Medium Income		High Income	Medium Income		High Income	Medium Income
Male	6	4	Family	1	3	Small	3	2
Female	4	6	Luxury	1	6	Medium	2	4
			Sports	8	1	Large	5	4

Feature Selection using Information Gain- Example (Solution)

$$\text{Entropy (Sample)} = -\left(\frac{10}{20} log_2 \frac{10}{20} + \frac{10}{20} log_2 \frac{10}{20}\right) = 1$$

Feature Selection using Information Gain- Example (Solution)

	C0	C1	Total
FAMILY	1	3	4
LUXORY	1	6	7
SPORTS	8	1	9
Total	10	10	20

Entropy(Sample, Family)

$$= -\left(\frac{1}{4} log_2 \frac{1}{4} + \frac{3}{4} log_2 \frac{3}{4}\right) = -\left(0.25 \times (-2) + 0.75 \times (-0.415)\right)$$

$$= 0.5 + 0.31125 = 0.81125$$

Entropy(Sample, Luxory)

$$= -\left(\frac{1}{7} log_2 \frac{1}{7} + \frac{6}{7} log_2 \frac{6}{7}\right) = -\left(\frac{1}{7} \times (-2.8074) + \frac{6}{7} \times (-0.2224)\right) = 0.401057 + 0.190628 = 0.591685$$

Entropy(Sample, Sports)

$$= -\left(\frac{8}{9} log_2 \frac{8}{9} + \frac{1}{9} log_2 \frac{1}{9}\right) = -\left(\frac{8}{9} \times (-0.16993) + \frac{1}{9} \times (-3.17)\right) = 0.151048 + 0.352222 = 0.50327$$

$$Average\ Information\ Entropy\ (Sample, Car\_Type) = \frac{4}{20} \times 0.8112 + \frac{7}{20} \times 0.5917 + \frac{9}{20} \times 0.5033 =$$

$$0.1622 + 0.2071 + 0.2265 = 0.5958$$

Feature Selection using Information Gain- Example (Solution)

Entropy(Sample, Male)=Entropy(Sample, Female)

$= -\left(\frac{6}{10} log_2 \frac{6}{10} + \frac{4}{10} log_2 \frac{4}{10}\right)$

$= -\left(0.6 \times (-0.737) + 0.4 \times (-1.322)\right)$

$= 0.4422 + 0.5288 = 0.971$

	C0	C1	Total
MALE	6	4	10
FEMALE	4	6	10
Total	10	10	20

$$Average\ Information\ Entropy\ (Sample, Gender) = \frac{10}{20} \times 0.971 + \frac{10}{20} \times 0.971 = 0.971$$

$$Information\ Gain\ (Sample, Gender) = 1 - 0.971 = 0.029$$

Feature Selection using Information Gain- Example (Solution)

	C0	C1	Total
SMALL	3	2	5
MEDIUM	2	4	6
LARGE	5	4	9
Total	10	10	20

Entropy(Sample, Small)

$= -\left(\frac{3}{5} log_2 \frac{3}{5} + \frac{2}{5} log_2 \frac{2}{5}\right) = -(0.6 \times (-0.737) + 0.4 \times (-1.322))$

$= 0.4422 + 0.5288 = 0.971$

Entropy(Sample, Medium)

$= -\left(\frac{2}{6} log_2 \frac{2}{6} + \frac{4}{6} log_2 \frac{4}{6}\right) = -\left(\frac{1}{3} \times (-1.5995) + \frac{2}{3} \times (-0.5778)\right) = 0.527835 + 0.0.381348 = 0.9092$

Entropy(Sample, Sports)

$= -\left(\frac{5}{9} log_2 \frac{5}{9} + \frac{4}{9} log_2 \frac{4}{9}\right) = -\left(\frac{5}{9} \times (-0.848) + \frac{4}{9} \times (-1.17)\right) = 0.47488 + 0.5148 = 0.9897$

$Average\ Information\ Entropy\ (Sample, House\_Type) = \frac{5}{20} \times 0.971 + \frac{6}{20} \times 0.9092 + \frac{9}{20} \times 0.9897 = 0.2426 + 0.2728 + 0.4454 = 0.9608$

$Information\ Gain\ (Sample, Shirt\_Type) = 1 - 0.9608 = 0.0392$

Car Type, House Type provide better Information Gain

Feature Selection using Gain Ratio

- Information Gain metric is biased towards a feature with large number of distinct values.
- This limitation of ID3 algorithm is handled by normalizing the *Information Gain* metric using a parameter called *SplitInfo*. The normalized Information Gain is called *Gain Ratio*.
- *Gain Ratio of an attribute A for a given dataset is computed as:*

$$Gain\ Ratio(S,A) = \frac{Information\ Gain(S,A)}{SplitInfo(S,A)}$$

$$Information\ Gain(S,A) = Entropy(S) - \sum_{v=1}^{|v|} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$SplitInfo(S,A) = -\sum_{v=1}^{|v|} \frac{|S_v|}{|S|} log_2\left(\frac{|S_v|}{|S|}\right)$$

where $\mathbf{S_v}$ is the set of rows in **S** for which the feature column **A** has value **v**, $|\mathbf{S_v}|$ is the number of rows in $\mathbf{S_v}$ and likewise **|S|** is the number of rows in **S.**

Importance of Split Info

Consider the following 3 situations, all with even distribution for simplicity of comparison:

- 10 possible outcomes with even distribution

  `SplitInfo = -10*(1/10*log2(1/10)) = 3.32`

- 100 possible outcomes with even distribution

  `SplitInfo = -100*(1/100*log2(1/100)) = 6.64`

- 1000 possible outcomes with even distribution

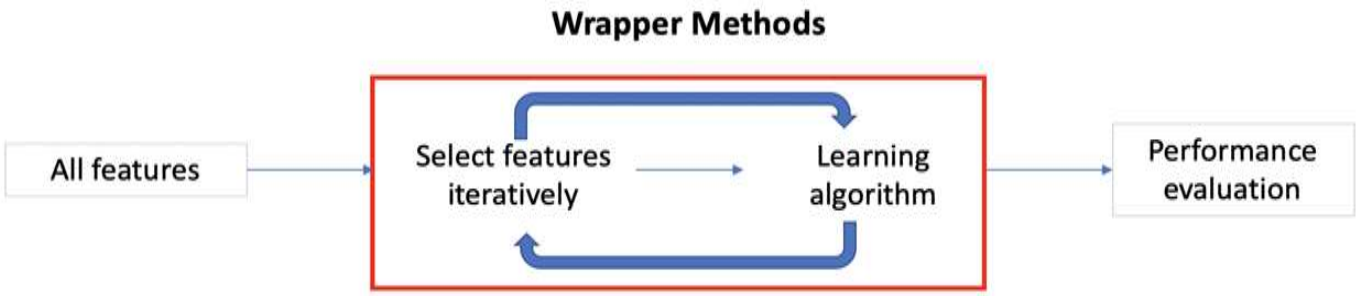
  `SplitInfo = -1000*(1/1000*log2(1/1000)) = 9.97`

So if you have to choose between 3 possible splitting scenarios, using only `Information Gain` as in ID3, the latter would be chosen. However, using `SplitInfo` in the `GainRatio`, it should be clear that as the number of choices goes **up**, the `SplitInfo` will also go up, and the `GainRatio` will go **down**.

Feature Selection Approaches

Wrapper Approach:

- In this approach, for every candidate subset, the learning model is trained and the result is evaluated by running the learning algorithm.
- Computationally very expensive but superior in performance.
- Requires some method to search the space of all possible subsets of features



Feature Selection Approaches

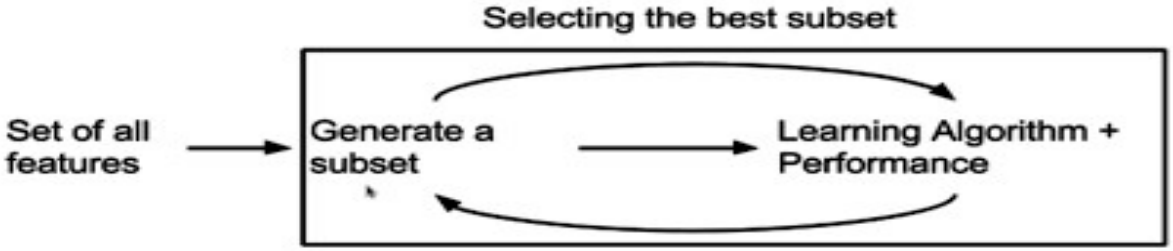
Wrapper Approach- Searching Methods:

- **Forward Feature Selection**
  - This is an iterative method wherein we start with the best performing variable against the target.
  - Next, we select another variable that gives the best performance in combination with the first selected variable.
  - This process continues until the preset criterion is achieved.
- **Backward Feature Elimination**
  - Here, we start with all the features available and build a model.
  - Next, we the variable from the model which gives the best evaluation measure value.
- **Exhaustive Feature Selection**
  - It tries every possible combination of the variables and returns the best performing subset.

Feature Selection Approaches

Embedded Approach

- These methods encompass the benefits of both the wrapper and filter methods.
- It includes interactions of features but also maintaining reasonable computational cost.
- Embedded methods are iterative in the sense that takes care of each iteration of the model training process and carefully extracts those features which contribute the most to the training for a particular iteration.



Embedded Approach

Random forests uses ***embedded approach*** to rank the importance of variables in a regression or classification problem in a natural way.

- The first step in measuring the feature importance in a data set is to fit a random forest to the data.
- During the fitting process the out-of-bag error for each data point is recorded and averaged over the forest.
- To measure the importance of the j-th feature after training, the values of the j-th feature are permuted among the training data and the out-of-bag error is again computed on this perturbed data set.
- The importance score for the j-th feature is computed by averaging the difference in out-of-bag error before and after all the permutation.
- The score is normalized by the variance of these differences.

Embedded Approach-Example

For a labelled classification dataset with 7 input features and an output variable, a random forest classifier has been trained. The Out-Of-Bag (OOB) score for the model is 0.905. In order to compute feature importance, each feature is permutated five times one-by-one and the OOB score is noted. Table 3 shows OOB score after each permutation for each feature. Compute the feature importance score of each feature and list three most important features for the model.

	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5	Feature 6	Feature 7
Permutation 1	0.7700	0.9037	0.7700	0.8987	0.9012	0.7462	0.8975
Permutation 2	0.7350	0.9000	0.7412	0.9100	0.9025	0.7375	0.9025
Permutation 3	0.7437	0.9075	0.7487	0.8987	0.9062	0.7465	0.8987
Permutation 4	0.7425	0.9012	0.7437	0.8987	0.9050	0.7162	0.9075
Permutation 5	0.7487	0.9012	0.7425	0.9062	0.9037	0.7450	0.8975

Embedded Approach-Example (Solution)

Features	Mean OOB Score After Permutation	OOB Score Before Permutation	d=|Difference between OOB score|	Squared Deviation of d from $\bar{d}$	Feature Score=d/Variance (d)
Feature 1	0.7480	0.905	0.157	0.0076	3.685446
Feature 2	0.9027	0.905	0.0023	0.0046	0.053991
Feature 3	0.7492	0.905	0.1558	0.0074	3.657277
Feature 4	0.9025	0.905	0.0025	0.0046	0.058685
Feature 5	0.9037	0.905	0.0013	0.0047	0.030516
Feature 6	0.7383	0.905	0.1667	0.0094	3.913146
Feature 7	0.9007	0.905	0.0043	0.0043	0.100939
			$\bar{d} = 0.0700$	Variance(d)=0.0426	

Top three important features are Feature 6, Feature 1, Feature 3.