What is Feature Selection ?

"Feature Selection is a process of selection a subset of Relevant Features(Variables or Predictors) from all features, which is used to make Model Building."

With N(high Dimension) number of features data analysis is challenging to the engineers in the field of Machine Learning and Data Mining.Feature Selection gives an effective way to solve this problem by removing irrelevant and redundant data, which can reduce computation time, improve learning accuracy, and facilitate a better understanding for the learning model or data.

How many Features to have in the Model?

One important thing is we have to take consideration Trade off between Predictive accuracy vs Model Interpretability. because if we use large number of Features the Predictive accuracy is likely to go up and Model Interpretability goes down.

If we have **less number of Features** then it is **easy to interpret the model**, **less likely to overfit** but it will give **low prediction accuracy**.

And if we have **large number of Features** then it is **difficult to Interpret model**, **more likely to overfit** and it will give **high prediction accuracy**.

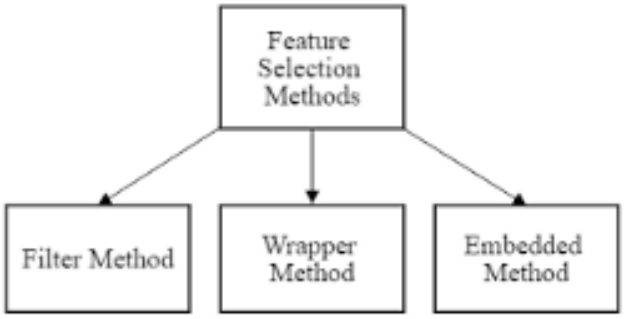
So below mention article has some ways to selection the number of features for the model.

Types of Feature Selection.

High number of features in the data increases the risk of **Overfitting** in the Model.

Feature Selection method helps to reduce the dimension of features by without much loss of information.

In this article, below are the some methods used for Feature Selection.



Feature Selection Methods

Filter Method.

Filter methods are also called as Single Factor Analysis. Using this method, the predictive power of each individual variable (feature) is evaluated. Various statistical means can be used to determine predictive power. One way is by correlating the feature with the target (what we are predicting). The features with the highest correlation are the best. Another way to determine predictive power is by determining the predictive (or information) value of the feature.

For Example : *Y* is target variable and (*X1, X2, X3,…Xn*) are independent variables. we find out the correlation between target variable with respect to independent variables. *(Y→ X1) ,(Y →X2),(Y →X3),…(Y →Xn). So the features which has highest Correlation Feature Selection(CFS) with Y we select that as a best features.*

2. Wrapper Method.

Wrapper methods use combinations of variables to determine predictive power. Common wrapper methods include: Subset Selection, Forward Stepwise, and Backward Stepwise(RFE). The wrapper method will find the best combination of variables. The wrapper method actually tests each feature against test models that it builds with them to evaluate the results. Out of all three methods, this is very computationally intensive. It is not recommended that this method be used on a high number of features.

a. Subset Selection.

In Subset selection we fits the model with each possible combinations of *N* features.

let say we have N numbers of independent Predictors (features) in a dataset, so we have total number of models in subset selection will be 2^*N* models. let say we have *N* = 2 (let say *X1* and *X2*). so we will have 2^2= 4 models .

*(Y = B0 , Y = B0 + B1*X1 , Y = C0 + C1*X2 , Y = D0 + D1*X1 + D2*X2)*

Subset Selection requires massive computational power for execute, suppose *N* = 10 then total models will be 2^{10} = 1024 models. To reduce this computational power it is divided into 2 parts.

Part 1 → fit all combination of models that has only *k* predictors out of total *N* predictors. Pick the best model from the set of all *k* predictors models (Model(*k*)). let say we have 4 predictors *(X1, X2, X3,X4)* i.e. *N* = 4.

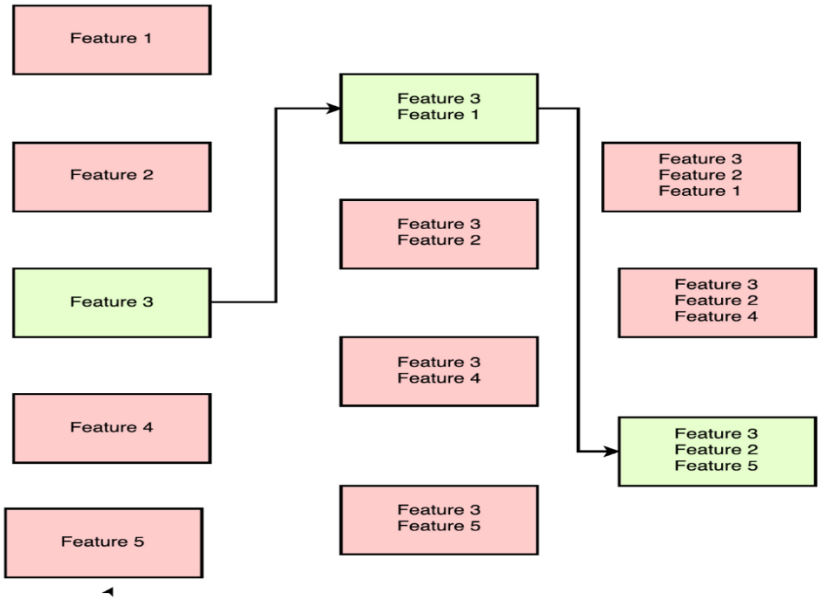
let say *k* = 1 then we will have 4 models i.e. *(Y = f(X1), Y = f(X2),Y = f(X3),Y = f(X4))*. we compute this 4 models and select best model out of them.

Now, let say *k* =2 then we will have 6 models i.e. *(Y = f(X1,X2) ,Y = f(X1,X3), Y = f(X1,X4),Y = f(X2,X3),Y = f(X2,X4) , Y = f(X3,X4))*.we compute this 6 models and select best model out of them. and we do so on for *k* values.

Part 2 → Select one best model from the *k* models i.e. Model(1), Model(2)…Model(*N*). to select best model we used (RSS(Residual Sum of Squares) , Cross Validation error or Adjusted R Square).

Note → Use Test error to evaluate the best features, otherwise if we use Training error for selection we might be end up selecting the model that has exactly N features.

b. Forward Stepwise selection.



Forward Selection method when used to select the best 3 features out of 5 features, Feature 3, 2 and 5 as the best subset.

Forward Stepwise selection initially starts with null model.i.e. starts with on variable in the model.

Then we add predictors (Features) one at Time and choose the best model among the bests of each *k* based on RSS,CV or adjusted R square.

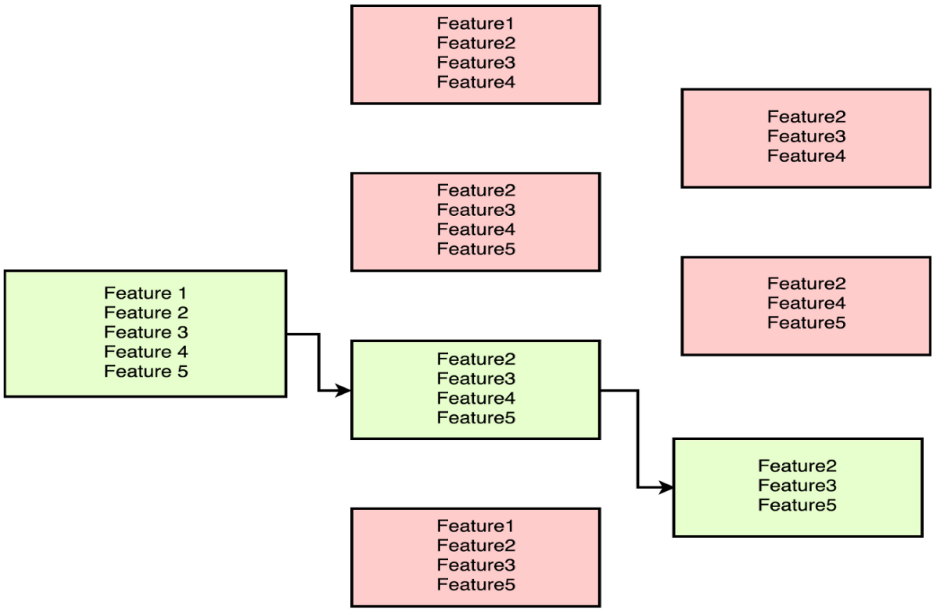
In this method once the predictor is selected it never drops in second step.

This is repeated till a best subset of 'k' predictors (features) are selected.

In forward selection, selection is the constrained as a predictor that is in model never drops. so selection models in forward selection becomes $1+N(N+1)/2$.

as we seen above, when N=10 in subset selection the total models was 1024, but in forward selection it reduces the computational power i.e. total model in this method will be 211.

c. Backward Stepwise Selection (Recursive Feature Elimination).



Recursive elimination eliminates the least explaining features one after the other. Feature 2,3 and 5 are the best subset of features arrived by Recursive elimination.

Backward selection works in the opposite direction in that it eliminates features. Because they are not run on every combination of features they are orders of magnitude less computationally intensive than straight subset selection.

Basically, it is opposite of Forward Stepwise selection. it starts with all Predictors and then drop one predictor at time and then select the best model.

Also, computational power is very similar to Forward Selection.

3. Embedded Method (Shrinkage).

Embedded Method is inbuilt variable selection method. we don't select or reject the predictors or variables in this method. this controls the value of parameters i.e. not so important predictors are given very low weight(close to zero), this is also know as Regularization.

a. LASSO Regression, the method which regularize the estimates or shrink the co-efficients of predictors to zero. in Lasso some of the co-efficients tends equal to zero *(β = 0)*. that why we drop or reject such predictors which gives *(β = 0)* .

$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{j=1}^{p}|\beta_j| \;=\mathrm{RSS}+\lambda\sum_{j=1}^{p}|\beta_j|.$$

b. RIDGE Regression, this adds a penalty, which equals the square of the magnitude of coefficients. All coefficients are shrunk by the same factor (so none of predictors are eliminated).

Ridge regression is very similar to least squares, except that the coefficients are estimated by minimizing a slightly different quantity. In particular, the ridge regression coefficient estimates $\hat{\beta}^R$ are the values that minimize

$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{j=1}^{p}\beta_j^2=\mathrm{RSS}+\lambda\sum_{j=1}^{p}\beta_j^2,$$

where $\lambda \geq 0$ is a *tuning parameter*, to be determined separately.

A tuning parameter (λ) controls the strength of the penalty term. When λ = 0, ridge regression equals least squares regression. If λ = ∞, all coefficients are shrunk to zero. The ideal penalty is therefore somewhere in between 0 and ∞.

Note → there is no guaranty that feature selection will lead to higher performance, it might work or not.if predictors are equally relevant for the problem then removing or rejecting predictors will be harmful.

These are some Feature Selection techniques, there are many more ways to do the feature selection.

Hope you like this article. Happy learning!!!!