**Variable Importance**

**Introduction:**

Variable selection is an important aspect of model building. It helps in building predictive models free from correlated variables, biases and unwanted noise. It helps in selecting a subset of relevant features (variables, predictors) for use in model construction and subset of a learning algorithm’s input variables upon which it should focus attention, while ignoring the rest

A lot of novice analysts assume that keeping all (or more) variables will result in the best model as you are not losing any information. Sadly, that is not true!

It happened many times that removing a variable from model has increased model accuracy. Such variables are often found to be correlated and hinder achieving higher model accuracy.

**Why variable selection??**

Actually the success of all Machine Learning algorithms depends on how you present the data. The features in your data will directly influence the predictive models you use and the results you can achieve. You can say that: the better the features that you prepare and choose, the better the results you will achieve.

When your goal is to get the best possible results from a predictive model, you need to get the most from what you have. This includes getting the best results from the algorithms you are using. It also involves getting the most out of the data for your algorithms to work with.

Many explored domains have hundreds to tens of thousands of variables/features with many irrelevant and redundant ones. In domains with many features the underlying probability distribution can be very complex and very hard to estimate (e.g. dependencies between variables). Irrelevant and redundant features can “confuse “learners.

There are multiple methods to select features and do feature engineering

1. Correlation method
2. Principal component analysis
3. Singular value decomposition
4. Factor Analysis
5. Boruta Algorithm
6. Machine learning algorithms like Decision trees, random forest.

Let us discuss popular methods. Correlation, we have already discussed in statistics module

**Boruta Algorithm:** Boruta is a feature selection algorithm. Precisely, it works as a wrapper algorithm around Random Forest. This algorithm derives its name from a demon in Slavic mythology that dwelled in pine forests.

Below is the step wise working of Boruta algorithm:

1. Firstly, it adds randomness to the given data set by creating shuffled copies of all features (which are called shadow features).
2. Then, it trains a random forest classifier on the extended data set and applies a feature importance measure (the default is Mean Decrease Accuracy) to evaluate the importance of each feature where higher means more important.
3. At every iteration, it checks whether a real feature has a higher importance than the best of its shadow features (i.e. whether the feature has a higher Z score than the maximum Z score of its shadow features) and constantly removes features which are deemed highly unimportant.
4. Finally, the algorithm stops either when all features gets confirmed or rejected or it reaches a specified limit of random forest runs.

Package “Boruta” will help you in R to implement above algorithm

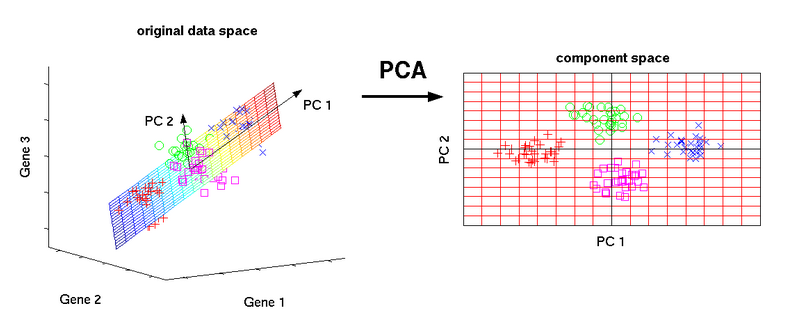
**Principle Component Analysis:** Principal component analysis is a method of extracting important variables (in form of components) from a large set of variables available in a data set. It extracts low dimensional set of features from a high dimensional data set with a motive to capture as much information as possible. With fewer variables, visualization also become much more meaningful. PCA is more useful when dealing with 3 or higher dimensional data.

It is always performed on a symmetric correlation or covariance matrix. This means the matrix should be numeric and have standardized data. Let’s understand it using an example:

Let’s say we have a data set of dimension 300 (n) × 50 (p). n represents the number of observations and p represents number of predictors. Since we have a large p = 50, there can be p(p-1)/2 scatter plots i.e more than 1000 plots possible to analyze the variable relationship. Wouldn’t is be a tedious job to perform exploratory analysis on this data?

In this case, it would be a lucid approach to select a subset of p (p << 50) predictor which captures as much information. Followed by plotting the observation in the resultant low dimensional space.

The image below shows the transformation of a high dimensional data (3 dimension) to low dimensional data (2 dimension) using PCA. Not to forget, each resultant dimension is a linear combination of p features



**Additional Information:**

Sometime we get confused between two fancy words Feature engineering and feature selection. Feature engineering means deriving new variables from the raw data and feature selection means selecting few variables from existing data.

Feature engineering is the process of transforming raw data into features that better represent the underlying problem to the predictive models, resulting in improved model accuracy on unseen data.

**Interview Questions:**

1. How will you deal with highly correlated variables?
2. Explain PCA & for 10 variables how many principle components will be generated by algorithm?
3. What is dimension reduction in Machine Learning?
4. What is curse of dimensionality?