Data Reduction in Data Mining

The method of data reduction may achieve a condensed description of the original data which is much smaller in quantity but keeps the quality of the original data.

Methods of data reduction:

Reducing Rows

- **Missing value removal**
- **Data Cube Aggregation**

Reducing Columns

- **Feature reduction (Dimension reduction)**
- **Feature selection (Feature extraction)**
- **Data Compression**
- **Numerosity Reduction**
- **Discretization & Concept Hierarchy Operation**

Reduction in Rows

1. **Data Cube Aggregation: (Reduction in Rows)**
   - This technique is used to aggregate data in a simpler form.
   - For example, imagine that information you gathered for your analysis for the years 2012 to 2014, that data includes the revenue of your company every three months. They involve you in the annual sales, rather than the quarterly average,
   - So we can summarize the data in such a way that the resulting data summarizes the total sales per year instead of per quarter. It summarizes the data.
2. **Missing Values Ratio**.
   - Data columns with too many missing values are unlikely to carry much useful information. Thus data columns with number of missing values greater than a given threshold can be removed. The higher the threshold, the more aggressive the reduction.

Reduction of Dimensions:

- Whenever we come across any data which is weakly important, then we use the attribute required for our analysis. It reduces data size as it eliminates outdated or redundant features.

An intuitive example of dimensionality reduction can be discussed through a simple e-mail classification problem, where we need to classify whether the e-mail is spam or not.

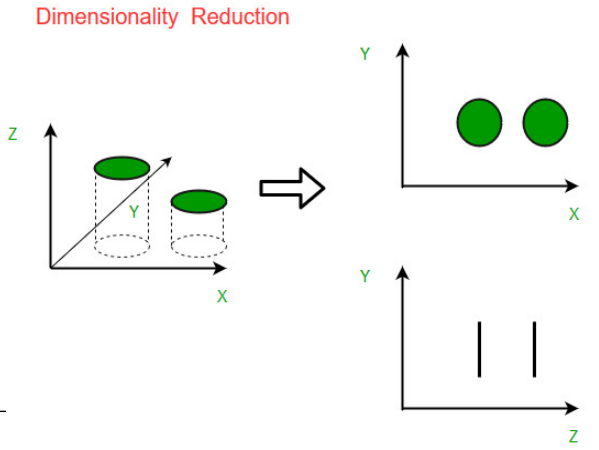
- This can involve a large number of features, such as whether or not the e-mail has a generic title, the content of the e-mail, whether the e-mail uses a template, etc.

- However, some of these features may overlap.

In another condition, a classification problem that relies on both humidity and rainfall can be collapsed into just one underlying feature, since both of the aforementioned are correlated to a high degree. Hence, we can reduce the number of features in such problems.

A 3-D classification problem can be hard to visualize, whereas a 2-D one can be mapped to a simple 2 dimensional space, and a 1-D problem to a simple line.

The below figure illustrates this concept, where a 3-D feature space is split into two 1-D feature spaces, and later, if found to be correlated, the number of features can be reduced even further.



Feature Selection Can be :

i**. Univariate Feature Selection**
ii Multi Variate

U***nivariate Feature Selection:*** *This technique involves more of a manual kind of work. Visiting every feature and checking its importance with the target. There are some great tricks that you should keep under your sleeve to implement* Univariate Feature Selection.

→ proper **domain knowledge** required ‘

→ **Checking the variance** (yes the ever confusing bias-variance trade-off :) of all features. The thumb rule here is set a threshold value (say a feature with 0 variance means it has the same value for every sample so such a feature would not bring any predictive power to the model) remove feature accordingly.

→ **Use of Pearson Correlation:** It might be the most applicable technique of three.

So, in a nutshell, it gives us the inter-dependence between the target variable and a feature.

When you have a lot of features (like hundreds or thousands of them), then it really becomes impossible to go & manually check for every one of them or if you don’t have enough domain knowledge then you got to trust this following technique. So put in layman’s term it is nothing but selecting multiple features at once.

Multivariate Feature Reduction

Feature Reduction (Extracting New feature):

This reduces the data in a high dimensional space to a lower dimension space, i.e. a space with lesser no. of dimensions

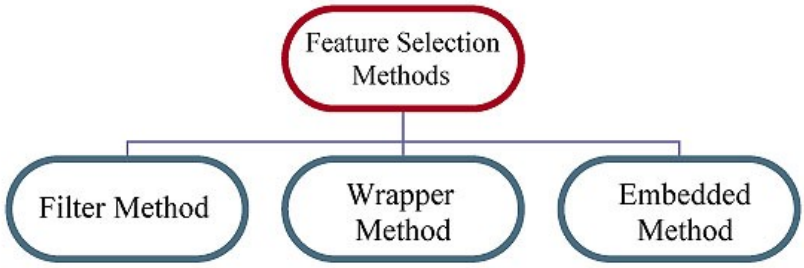
The various methods used for dimensionality reduction include

- Principal Component Analysis (PCA)
- Linear Discriminant Analysis (LDA)
- Generalized Discriminant Analysis (GDA)

Multivariate Feature Selection:

In this, we try to find a subset of the original set of variables, or features, to get a smaller subset which can be used to model the problem.

Multivariate Feature Selection is broadly divided into three categories:



Filter Method :

→ Filter methods are generally used as a preprocessing step. The selection of features is independent of any machine learning algorithms.

→ Filter methods apply some ranking over features. The ranking denotes how 'useful' each feature is likely to be for classification. Once this ranking has been computed, a feature set composed of the best N features is created.

→ Features are selected on the basis of their scores in various statistical tests for their correlation with the outcome variable. (The correlation is a subjective term here).

1. **Pearson's Correlation:** Oh yes! Pearson Correlation is Filter method. We have already discussed it.
2. **Variance Thresholds:** This too, we have already discussed.
3. **Linear discriminant analysis (LDA):** The goal is to project a dataset onto a lower-dimensional space with good class-separability and also reduce computational costs. In simple word LDA brings all the higher dimensional variable (which we can't plot and analyse) onto 2D graph & while doing so removes the useless feature.

→ LDA is also 'S***upervised Dimension Reduction'*** technique and more kind of **Feature Extraction** than **Selection** (as it is creating kind of a new variable by reducing its dimension). So it works only on labelled data.

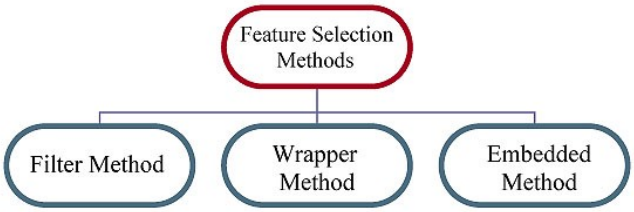
→ It maximizes the *separability* between classes.

4. **ANOVA (Analysis of variance)** It is similar to LDA except for the fact that it is operated using one or more categorical independent features and one continuous dependent feature (Response or target). It provides a statistical test of whether the means of several groups are equal or not.

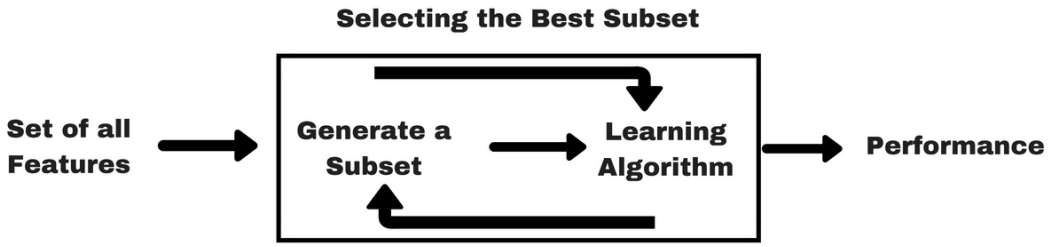
5. **Chi-Square:** It is a is a statistical test applied to the groups of categorical features to evaluate the likelihood of correlation or association between them using their frequency distribution.

Note : filter method do not remove multi-collinearity. So, you must deal with multi-collinearity of features as well before training models for your data.

Feature\Response	Continuous	Categorical
Continuous	Pearson's Correlation	LDA
Categorical	Anova	Chi-Square



Wrapper Method:



- Based on the inferences that we draw from the previous model, we decide to add or remove features from your subset.
- Wrapper methods are so called because they wrap a classifier up in a feature selection algorithm. Typically a set of features is chosen; the efficiency of this set is determined; some perturbation is made to change the original set and the efficiency of the new set is evaluated.
- The problem with this approach is that feature space is vast and looking at every possible combination would take a large amount of time and computation.
- The problem is essentially reduced to a search problem. These methods are usually computationally very expensive.

1. **Forward Selection:** Forward selection is an iterative method in which we start with having no feature in the model. In each iteration, we keep adding the feature which best improves our model till an addition of a new variable does not improve the performance of the model.
2. **Backward Elimination:** In backward elimination, we start with all the features and removes the least significant feature at each iteration which improves the performance of the model. We repeat this until no improvement is observed on the removal of features.
3. **Recursive Feature Elimination (RFE):** It works by recursively removing attributes and building a model on those attributes that remain. It uses an external estimator that assigns weights to features (for example, the coefficients of a linear model) to identify which attributes (and the combination of attributes) contribute the most to predicting the target attribute.
   → It is a greedy optimization algorithm which aims to find the best performing feature subset.
   → It repeatedly creates models and keeps aside the best or the worst performing feature at each iteration.
   → It constructs the next model with the left features until all the features are exhausted. It then ranks the features based on the order ofheir elimination

Embbeded Method:

→ Embedded methods combine the qualities' of filter and wrapper methods. It's implemented by algorithms that have their own built-in feature selection methods.

→ So this are not any kind of special feature selection or extraction techniques and they also help in avoiding overfitting.

1. Select k-best in Random Forest
2. Gradient boosting machine (GBM)

Difference between Filter and Wrapper method

Filter Method	Wrapper Method
Measure the relevance of features by their correlation with dependent variable	Measure the usefulness of a subset of feature by actually training a model on it.
Much faster compared to wrapper methods as they do not involve training the models	Wrapper methods are computationally very as they involve training the models
Use statistical methods for evaluation of a subset of features	Wrapper methods use cross validation.

3. **Data Compression:**

- The data compression technique reduces the size of the files using different encoding mechanisms (Huffman Encoding & run-length Encoding). We can divide it into two types based on their compression techniques.
    - **Lossless Compression –**
      Encoding techniques (Run Length Encoding) allows a simple and minimal data size reduction. Lossless data compression uses algorithms to restore the precise original data from the compressed data.

    - **Lossy Compression –**
      Methods such as Discrete Wavelet transform technique, PCA (principal component analysis) are examples of this compression. For e.g., JPEG image format is a lossy compression, but we can find the meaning equivalent to the original the image. In lossy-data compression, the decompressed data may differ to the original data but are useful enough to retrieve information from them.

4. **Numerosity Reduction:**

- In this reduction technique the actual data is replaced with mathematical models or smaller representation of the data instead of actual data, it is important to only store the model parameter.
- Or non-parametric method such as clustering, histogram, sampling.

5. **Discretization & Concept Hierarchy Operation:**

- Techniques of data discretization are used to divide the attributes of the continuous nature into data with intervals.
- We replace many constant values of the attributes by labels of small intervals. This means that mining results are shown in a concise, and easily understandable way.
    - **Top-down discretization –**
      If you first consider one or a couple of points (so-called breakpoints or split points) to divide the whole set of attributes and repeat of this method up to the end, then the process is known as top-down discretization also known as splitting.
    - **Bottom-up discretization –**
      If you first consider all the constant values as split-points, some are discarded through a combination of the neighbourhood values in the interval, that process is called bottom-up discretization.

- Concept Hierarchies:

    - It reduces the data size by collecting and then replacing the low-level concepts (such as 43 for age) to high-level concepts (categorical variables such as middle age or Senior).

Some interesting Feature reduction techniques are

- **Low Variance Filter**.

- data columns with little changes in the data carry little information.
- Thus all data columns with variance lower than a given threshold are removed.
- Note: variance is range dependent; therefore normalization is required before applying this technique.

- **High Correlation Filter.**

  - Data columns with very similar trends are also likely to carry very similar information.
  - In this case, only one of them will suffice to feed the machine learning model.
  - Here we calculate the correlation coefficient between numerical columns and between nominal columns as the Pearson's Product Moment Coefficient and the Pearson's chi square value respectively.
  - Pairs of columns with correlation coefficient higher than a threshold are reduced to only one. A word of caution: correlation is scale sensitive; therefore column normalization is required for a meaningful correlation comparison.

Random Forests / Ensemble Trees.

- Decision Tree Ensembles, also referred to as random forests, are useful for feature selection in addition to being effective classifiers.
- One approach to dimensionality reduction is to generate a large and carefully constructed set of trees against a target attribute and then use each attribute's usage statistics to find the most informative subset of features.
- Specifically, we can generate a large set (2000) of very shallow trees (2 levels), with each tree being trained on a small fraction (3) of the total number of attributes.
- If an attribute is often selected as best split, it is most likely an informative feature to retain. A score calculated on the attribute usage statistics in the random forest tells us – relative to the other attributes – which are the most predictive attributes.

Principal Component Analysis (PCA).

- Principal Component Analysis (PCA) is a statistical procedure that orthogonally transforms the original *n* coordinates of a data set into a new set of *n* coordinates called principal components.
- As a result of the transformation, the first principal component has the largest possible variance; each succeeding component has the highest possible variance under

the constraint that it is orthogonal to (i.e., uncorrelated with) the preceding components.

- Keeping only the first $m < n$ components reduces the data dimensionality while retaining most of the data information, i.e. the variation in the data.
- Notice that the PCA transformation is sensitive to the relative scaling of the original variables. Data column ranges need to be normalized before applying PCA.
- Also notice that the new coordinates (PCs) are not real system-produced variables anymore. Applying PCA to your data set loses its interpretability. If interpretability of the results is important for your analysis, PCA is not the transformation for your project

- **Backward Feature Elimination**. In this technique, at a given iteration, the selected classification algorithm is trained on *n* input features. Then we remove one input feature at a time and train the same model on *n-1* input features *n* times. The input feature whose removal has produced the smallest increase in the error rate is removed, leaving us with *n-1* input features. The classification is then repeated using *n-2* features, and so on. Each iteration *k* produces a model trained on *n-k* features and an error rate *e(k)*. Selecting the maximum tolerable error rate, we define the smallest number of features necessary to reach that classification performance with the selected machine learning algorithm.

- **Forward Feature Construction**. This is the inverse process to the Backward Feature Elimination. We start with 1 feature only, progressively adding 1 feature at a time, i.e. the feature that produces the highest increase in performance. Both algorithms, Backward Feature Elimination and Forward Feature Construction, are quite time and computationally expensive. They are practically only applicable to a data set with an already relatively low number of input columns.

  -

For reference

- **Step-wise Forward Selection –**
  The selection begins with an empty set of attributes later on we decide best of the original attributes on the set based on their relevance to other attributes. We know it as a p-value in statistics.

Suppose there are the following attributes in the data set in which few attributes are redundant.

Initial attribute Set: {X1, X2, X3, X4, X5, X6}

Initial reduced attribute set: { }

Step-1: {X1}

Step-2: {X1, X2}

Step-3: {X1, X2, X5}

Final reduced attribute set: {X1, X2, X5}

- **Step-wise Backward Selection –**
  This selection starts with a set of complete attributes in the original data and at each point, it eliminates the worst remaining attribute in the set.

Suppose there are the following attributes in the data set in which few attributes are redundant.

Initial attribute Set: {X1, X2, X3, X4, X5, X6}

Initial reduced attribute set: {X1, X2, X3, X4, X5, X6 }

Step-1: {X1, X2, X3, X4, X5}

Step-2: {X1, X2, X3, X5}

Step-3: {X1, X2, X5}

Final reduced attribute set: {X1, X2, X5}

- **Combination of forwarding and Backward Selection –**

It allows us to remove the worst and select best attributes, saving time and making the process faster.