**4. Statistical Modeling and Machine Learning**

**4.1. Introduction to model selection**

**Model selection** is the task of **selecting** a statistical **model** from a set of models and apply on the given **data**. Pre-existing set of **data** is considered. The task also involve the design of experiments such that the **data**collected is well-suited to the problem of **model selection**.

**Overfitting :**

**overfitting** is "the production of an analysis that corresponds too closely or exactly to a particular set of data, and may therefore fail to fit additional data or predict future observations reliably".

**Underfitting :**

**Underfitting** occurs when a statistical model cannot adequately capture the underlying structure of the data. An **underfitted model** is a model where some parameters or terms that would appear in a correctly specified model are missing. Underfitting would occur, for example, when fitting a linear model to non-linear data. Such a model will tend to have poor predictive performance.

Overfitting and underfitting can occur in [machine learning](https://en.wikipedia.org/wiki/Machine_learning)

To prevent overfitting and underfitting we have models like

* Regularization
* Bias/variance tradeoff (e.g. parsimony)
* AIC
* BIC
* Cross validation
* Ridge regression and penalized regression (e.g LASSO)

**Features of model selection :**

* **Accuracy**

Accuracy deals with the perfection in the solution from the given data set how well your algorithm provides an accurate prediction is considered here.

* **Training time**

Various Machine Learing algorithm works in different ways and the time needed to train the given data set is an important factors if the data set has too much of variance the machine will take time to learn the training data.

* **Linearity**

Many machine learning tries to use the linear classification technique where the data set is assumed to separated by the straight line.

* **Number of parameters**

It is a very important feature depending on number of parameters in the given data set the prediction can vary.

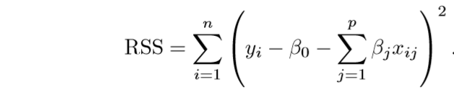
**4.1.1.Regularization**

*Regularizaton is a form of regression, that constrains or regularizes or shrinks the coefficient estimates towards zero. In other words,*this technique discourages learning a more complex or flexible model, so as to avoid the risk of overfitting.

A simple relation for linear regression looks like this. Here Y represents the learned relation and β represents the coefficient estimates for different variables or predictors(X).

**Y ≈ β0 + β1X1 + β2X2 + …+ βpXp**

The fitting procedure involves a loss function, known as residual sum of squares or RSS. The coefficients are chosen, such that they minimize this loss function.

**

Now, this will adjust the coefficients based on your training data. If there is noise in the training data, then the estimated coefficients won’t generalize well to the future data. This is where regularization comes in and shrinks or regularizes these learned estimates towards zero.

**4.1.2. Bias/variance tradeoff e.g. parsimony**

There are two fundamental causes of prediction error for a model -**bias** and **variance**.

It is a property of different predictive models in which models with a lower bias for parameter estimation have a higher variance for the given data set and vice versa.

**Bias** – It is an error form the erroneous assumption made during the learning of an algorithm. Higher bias can lead to missing of the relevant data of feature needed for the targeted value in other words it leads to underfitting.

**Variance** – It is an error form the sensitivity of an algorithm where small fluctuations of samples in the training set can lead to an error. High variance in an algorithm can lead to generation of random noise in the training data and can deviate the output. In other words it leads to overfitting.

**Bias-Variance tradeoff** is generally faced in supervised algorithms due to which the accuracy and generalization both cannot be adopted in the model. The goal of this is to achieve low bias and low variance in order to achieve good prediction performance.

Any model can be bad or not optimal because of two main reason :

1. It is not accurate
2. It does not match the data well.

The reason for the first is bias and other is variance. If models are made complex then it leads to improvement in the bias but such models are very costly which leads to higher variance, whereas when the model is made more specific to the data then the variance will be reduced but on the other hand it leads to higher bias.

For example, if a straight line and high degree polynomial curve is considered the straight line will have no variance at all but it will have bias since it is a bad fit line, where as the polynomial curve will have no bias as the curve can be fitted according to the point but this leads to high variance. Bias error is the simplifying assumptions made by a model to make the target function easier to learn.

Models with high bias are fast to learn and easier to understand but are less flexible, also they have lower predictive performance for the complex problems. Low bias leads to less assumption about the form of the target function where as high bias leads to more assumption about the form of target function.

Variance is the amount that the estimate of the target functions which will change if different training data was used. Algorithm should have some varience.

Low variance provides small changes to the estimate of the target function with the changes to the training dataset. High variance provides large changes to the estimate of the target function with changes to the training data set. Whenever model is choosing with low complexity and low variance is introduced.

**Principal of Parsimony**

Parsimony principle is used in bias variance trade off and it tell us that as we increase the number of parameters in a model bias goes on decreasing and variance keeps on increasing.

Parsimony principle underlines model selection approaches. It also tells that when the numbers of parameters are increased the variance of the system increases and vice versa.

**4.1.3. AIC**

AIC (Alkaline Information Criterion) is similar to a statistical model for estimating the given data set. It is a model selector and given number of models it estimates the quality of each model in comparison with other models and provides with the best model.

If any model is estimated on a particular set of data AIC scores will provide an estimation of that model performance for the new data set. AIC is similar to the in-sample error of the estimated model.

To select the model using AIC the model which provides smallest AIC value with respect to other is chosen. To avoid the risk of overfitting AIC provides penalty by the term 2\*d.

The term “2 x d” increases as the number of parameters increases and thus it reduces the complexity of the model. AIC is best suited with respect to BIC when the model is more complex.

Mathematical Foundation of AIC

AIC = - 2 (log – likelihood) + 2 \*d

Where d is the number of model parameters i.e. the number of variables and the intercept.

Log-likelihood is measure of model fit and it is obtained from statistical output.

**4.1.4.BIC**

Bayesian Information Criterion also known as Schwarz criterion is a criterion for selection of an appropriate model out of many available finite set of models. It is partially based on the likelihood function and is also related to Alkaline Information Criterion(AIC).

BIC is used to solve the problem of the overfitting faced by models it does so by introducing the penalty term of the number of parameters or features in the model. The value of penalty term in BIC is more than that used in AIC. BIC is used widely with the time series data set or linear regression.

**Mathematical Foundation of BIC**

BIC assumes that the data distribution is done in the exponential family. Following are various parameters and the formula for BIC :

x = the observed data;

n = number of data points in the observed data x or the sample size;

k = number of free parameters to be estimated if the estimated model is considered to

be linear regression then k here is the number of regressors and intercepts.

P(x|k) = It is the probability of the observed data for the given number of parameters or

in other words it is the likelihood of the parameters for the given dataset.

L = maximum value of Likelihood function for the estimated model.

**Formula for BIC**

ln .p(x|k) = BIC = -2 . ln . L + k . ln(n)

**Characteristics of BIC**

1. BIC is independent of the constant.
2. It is used to measure the efficiency of model consisting parameters in terms of the predicting data.
3. When there are large numbers of parameters BIC penalized these parameters and hence reduce the complexity of the model.
4. The value obtained after BIC is equal to the minimum description length of that criteria but it has a negative sign.
5. BIC can be used to choose number of cluster depending on the complexity present in the dataset.
6. It is similar to other penalized methods such as RIC and AIC.

**4.1.5. Cross Validation**

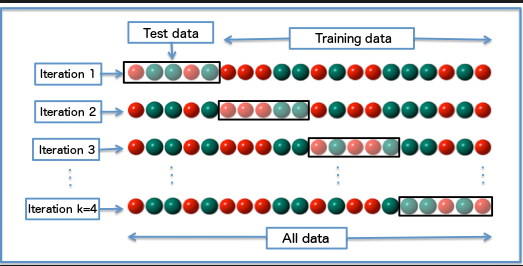
Cross validation (CV) is one of the technique used to test the effectiveness of a machine learning models, it is also a re-sampling procedure used to evaluate a model if we have a limited data.

It is a technique to evaluate predictive models by recursively making partitions of the original sample into a training set to train the model and a test set to evaluate it.

Cross Validation is a statistical method used to estimate the skill of machine learning models.

Cross Validation is also known as rotation estimation or out of sample testing. One round of cross validation involves portioning a sample of data into various subsets and performs the analysis on one subset also known as training set and validating the analysis on other subset known as the validation set of testing set.

In Order to reduce the variability many methods involves number of rounds of cross validation using different partitions and the validation results are combined to give an estimate to the model’s predictive performance.



From above figure it is clear how the input samples are selected for performing the training and testing in every iteration. The above method can be called as four fold cross validation as the process of cross validation is applied four times.

**4.1.6. Ridge regression**

Ridge expression is a technique which comes into picture when the data suffers from multicollinearity (which simply means that independent variables are highly correlated).

**Ridge regression is used to create a parsimonious model in the following scenarios:**

* The number of predictor variables in a given set exceeds the number of observations.
* The dataset has multicollinearity (correlations between predictor variables).

The term multicollinearity also refers to collinearity concept in statistics. In this phenomenon, one predicted value in multiple regression models is linearly predicted with others to attain a certain level of accuracy.

The concept multicollinearity occurs when there are high correlations between more than two predicted variables.

In multicollinearity concept, event though the least squares estimates(OLS) are unbiased, their variances are large which in turn return results in the deviation of the observed value far from the true value.

By adding a degree of bias to the regression estimates, ridge expression is able to reduce the standard errors.

The equation for linear regression

y = a + b \*x

Equation also has an error term. The complete equation becomes :

y = a + b \* x + e

Where e is the error term, it is the value needed to correct for a prediction error between the observed and predicted value.

Working of Ridge expression :

Ridge regression performs **L2 regularization**. Here the penalty equivalent is added to the square of the magnitude of coefficients. The minimization objective is as followed.

Taking a response **vector y ∈ Rn** and a predictor **matrix X ∈Rn×p,** the ridge **regression coefficients** are defined as:

* Here **λ** is the turning factor that controls the strength of the penalty term.
* If **λ = 0**, the objective becomes similar to simple linear regression. So we get the same coefficients as simple linear regression.
* If **λ = ∞**, the coefficients will be zero because of infinite weightage on the square of coefficients as anything less than zero makes the objective infinite.
* If **0 <λ<∞**, the magnitude of λ decides the weightage given to the different parts of the objective.
* In simple terms, the minimization objective = **LS Obj + λ** (sum of the square of coefficients)
* Where **LS Obj** is Least Square Objective that is the linear regression objective without regularization.

As **ridge regression in r** shrinks the coefficients towards zero, it introduces some bias. But it can reduce the variance to a great extent which will result in a better mean-squared error. The amount of shrinkage is controlled by λ which multiplies the ridge penalty. As large λ means more shrinkage, we can get different coefficient estimates for the different values of λ.

### Ridge Regression Example:

* Linear Regression
* Logistic Regression
* Polynomial Regression
* Stepwise Regression
* Lasso Regression
* Regression Analysis
* Multiple Regression

**4.1.7.Lasso Regression**

**Lasso regression** which is one of the regression models that are available to analyze the data.

**LASSO** stands for **Least Absolute Shrinkage and Selection Operator**.

Lasso regression is one of the regularization methods that creates parsimonious models in the presence of large number of features, where large means either of the below two things:

1. Large enough to enhance the tendency of the model to over-fit. Minimum ten variables can cause overfitting.

2. Large enough to cause computational challenges. This situation can arise in case of millions or billions of features.

Lasso regression performs L1 regularization that is it adds the penalty equivalent to the absolute value of the magnitude of the coefficients. Here the minimization objective is as followed.

Minimization objective = LS Obj + λ (sum of absolute value of coefficients)

Where LS Obj stands for Least Squares Objective which is nothing but the linear regression objective without regularization and λ is the turning factor that controls the amount of regularization. The bias will increase with the increasing value of λ and the variance will decrease as the amount of shrinkage (λ) increases.

Here the turning factor λ controls the strength of penalty, that is

1. When  λ = 0: We get same coefficients as simple linear regression
2. When λ = ∞: All coefficients are zero
3. When 0 <λ<∞: We get coefficients between 0 and that of simple linear regression

So when λ is in between the two extremes, we are balancing the below two ideas.

* 1. Fitting a linear model of y on X
* 2. Shrinking the coefficients

But the nature of L1 regularization penalty causes some coefficients to be shrunken to zero. Hence, unlike ridge regression, lasso regression is able to perform variable selection in the liner model. So as the value of λ increases, more coefficients will be set to value zero (provided fewer variables are selected) and so among the nonzero coefficients, more shrinkage will be employed.

**5. Data Transformation**

**5.1. What is Data Transformation?**

Data transformation is the process of converting data from one format or structure into another format or structure. Data transformation is critical to activities such as data integration and data management. Data transformation can include a range of activities: you might convert data types, cleanse data by removing nulls or duplicate data, enrich the data, or perform aggregations, depending on the needs of your project.

Typically, the process involves two stages.

In the first stage, you:

* Perform data discovery where you identify the sources and data types.
* Determine the structure and data transformations that need to occur.
* Perform data mapping to define how individual fields are mapped, modified, joined, filtered, and aggregated.

In the second stage, you:

* Extract data from the original source. The range of sources can vary, including structured sources, like databases, or streaming sources, such as telemetry from connected devices, or log files from customers using your web applications.
* Perform transformations. You transform the data, such as aggregating sales data or converting date formats, editing text strings, or joining rows and columns.
* Send the data to the target store. The target might be a database or a data warehouse that handles structured and unstructured data.

## Need To Transform Data?

You might want to transform your data for a number of reasons. Generally, businesses want to transform data to make it compatible with other data, move it to another system, join it with other data, or aggregate information in the data.

For example, consider the following scenario: your company has purchased a smaller company, and you need to combine information for the Human Resources departments. The purchased company uses a different database than the parent company, so you'll need to do some work to ensure that these records match. Each of the new employees has been issued an employee ID, so this can serve as a key. But, you'll

need to change the formatting for the dates, you'll need to remove any duplicate rows, and you'll have to ensure that there are no null values for the Employee ID field so that all employees are accounted for. All these critical functions are performed in a staging area before you load the data to the final target.

Other common reasons to transform data include:

* You are moving your data to a new data store; for example, you are moving to a cloud data warehouse and you need to change the data types.
* You want to join unstructured data or streaming data with structured data so you can analyze the data together.
* You want to add information to your data to enrich it, such as performing lookups, adding geolocation data, or adding timestamps.
* You want to perform aggregations, such as comparing sales data from different regions or totaling sales from different regions.

## How Is Data Transformed?

There are a few different ways to transform data:

* **Scripting.** Some companies perform data transformation via scripts using SQL or Python to write the code to extract and transform the data.
* **On-premise ETL tools**. ETL (Extract, Transform, Load) tools can take much of the pain out of scripting the transformations by automating the process. These tools are typically hosted on your company's site, and may require extensive expertise and infrastructure costs.
* **Cloud-based ETL tools**. These ETL tools are hosted in the cloud, where you can leverage the expertise and infrastructure of the vendor.

## Data Transformation Challenges

Data transformation can be difficult for a number of reasons:

* **Time-consuming**. You may need to extensively cleanse the data so you can transform or migrate it. This can be extremely time-consuming, and is a common complaint amongst data scientists working with unstructured data.
* **Costly**. Depending on your infrastructure, transforming your data may require a team of experts and substantial infrastructure costs.
* **Slow.** Because the process of extracting and transforming data can be a burden on your system, it is often done in batches, which means you may have to wait up to 24 hours for the next batch to be processed. This can cost you time in making business decisions.

## Some Data Transformation Strategies:-

### 1 Smoothing

Smoothing is a process of removing noise from the data.

### 2 Aggregation

Aggregation is a process where summary or aggregation operations are applied to the data.

### 3 Generalization

In generalization low-level data are replaced with high-level data by using concept hierarchies climbing.

### 4 Normalization

Normalization scaled attribute data so as to fall within a small specified range, such as 0.0 to 1.0.

### 5 Attribute Construction

In Attribute construction, new attributes are constructed from the given set of attributes.

**5.2. Dimension Reduction**

In machine learning classification problems, there are often too many factors on the basis of which the final classification is done. These factors are basically variables called features.

During classification, there may be case when we come across “N” number of dimensions or features/parameters/attributes.

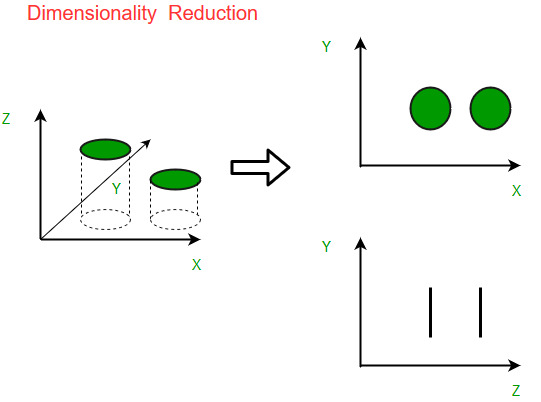
All these features many times will classify the data into wrong classes due to certain values of the number of feautes.

The motivation behind dimension reduction is to eliminate or remove such unwanted dimensions or features which will finally classify the dataset into correct class. More number of features can also lead to complex classification.

Dimensionlality reduction can also referred to the process of reducing a set of data having vase dimensions into the data with lesser dimensions ensuring that it provides the same or similar information.

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.



There are two components of dimensionality reduction:

* **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. It usually involves three ways:
  1. Filter
  2. Wrapper
  3. Embedded
* **Feature extraction:** This reduces the data in a high dimensional space to a lower dimension space, i.e. a space with lesser no. of dimensions. The best technique for feature extraction is Principal Component Analysis(PCA). PCA algorithm works with linear data set.

**Methods of Dimensionality Reduction**

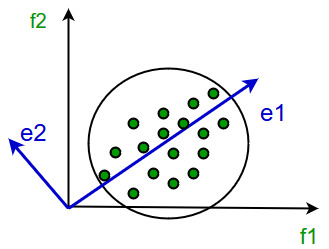
The various methods used for dimensionality reduction include:

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

Dimensionality reduction may be both linear or non-linear, depending upon the method used. The prime linear method, called Principal Component Analysis, or PCA, is discussed below.

**Principal Component Analysis**

This method was introduced by Karl Pearson. It works on a condition that while the data in a higher dimensional space is mapped to data in a lower dimension space, the variance of the data in the lower dimensional space should be maximum.



It involves the following steps:

* Construct the covariance matrix of the data.
* Compute the eigenvectors of this matrix.
* Eigenvectors corresponding to the largest eigenvalues are used to reconstruct a large fraction of variance of the original data.

Hence, we are left with a lesser number of eigenvectors, and there might have been some data loss in the process. But, the most important variances should be retained by the remaining eigenvectors.

There are few more dimensionality reduction techniques listed as follows:

**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.

**Low Variance Filter**. Similarly to the previous technique, data columns with little changes in the data carry little information. Thus all data columns with variance lower than a given threshold are removed. A word of caution: 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](http://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient) and the [Pearson’s chi square value](http://en.wikipedia.org/wiki/Pearson%27s_chi-squared_test) 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 trees or random forest trees are useful methods for the selection of the best features.

Set of sub-trees are made based on the target variable or feature and then the sub-tree providing highest gain for that target attribute is selected as the best classifier so out of “n” sub-trees formed with various features only one with the highest gaing is selected as the best classifier all others are removed which leads to reduction of many features.

**Backward Feature Elimination**.

In this technique initially all the features are considered as the input features and then one by one some features are removed and its effect is seen, if removal of any feature leads to increase in the gain then that feature is permanently removed for certain classification and if that feature leads to decrease in the gain that feature is again added back to the classifier.

Backward feature elimination is an iterative process, on every iteration it is checked for where to remove or keep the feature.

The process is ongoing and stops when maximum gain or beast classification is done.

**Forward Feature Construction**.

**This technique is completely reverse of the backward feature elimination. It has only one similarity that it is also iterative process.**

**Here initially only one feature is considered, on every iteration on more feature is added, after adding that feature if the system provides increase in gain than that feature is kept else that feature is eliminated.**

**Advantages of Dimensionality Reduction**

* It helps in data compression, and hence reduced storage space.
* It reduces computation time.
* It also helps remove redundant features, if any.

**Disadvantages of Dimensionality Reduction**

* It may lead to some amount of data loss.
* PCA tends to find linear correlations between variables, which is sometimes undesirable.
* PCA fails in cases where mean and covariance are not enough to define datasets.
* We may not know how many principal components to keep- in practice, some thumb rules are applied.

**5.3. Feature Extraction**

Feature extraction is a process of dimensionality reduction by which an initial set of raw data is reduced to more manageable groups for processing. A characteristic of these large data sets is a large number of variables that require a lot of computing resources to process. Feature extraction is the name for methods that select and /or combine variables into features, effectively reducing the amount of data that must be processed, while still accurately and completely describing the original data set. The best technique for feature extraction is Principal Component Analysis (PCA). PCA algorithm works with linear data set.

### Why feature extraction is Useful?

The process of feature extraction is useful when you need to reduce the number of resources needed for processing without losing important or relevant information. Feature extraction can also reduce the amount of redundant data for a given analysis. Also, the reduction of the data and the machine’s efforts in building variable combinations (features) facilitate the speed of learning and generalization steps in the [machine learning](https://deepai.org/machine-learning-glossary-and-terms/machine-learning) process.

### Practical Uses of Feature Extraction

* **Autoencoders**

– The purpose of [autoencoders](https://deepai.org/machine-learning-glossary-and-terms/autoencoder) is [unsupervised learning](https://deepai.org/machine-learning-glossary-and-terms/unsupervised-learning) of efficient data coding. Feature extraction is used here to identify key features in the data for coding by learning from the coding of the original data set to derive new ones.

* **Bag-of-Words**

– A technique for [natural language processing](https://deepai.org/machine-learning-glossary-and-terms/natural-language-processing) that extracts the words (features) used in a sentence, document, website, etc. and [classifies](https://deepai.org/machine-learning-glossary-and-terms/classifier) them by frequency of use. This technique can also be applied to image processing.

* **Image Processing** – Algorithms are used to detect features such as shaped, edges, or motion in a digital image or video.

**5.4. Smoothing and Aggregating**

**5.4.1. Smoothing**

Data smoothing is done by using an algorithm to remove noise from a data set.

Noise is unwanted quantity in signal and it is random. Various techniques included in smooting are as follows :

**1. Binnig mehods**

In Binning method to smooth the data, firstly the data is needed to be sorted and for sorting the data the neighborhood values or the surrounded values are compared.

The sorted values are then distributed into number of buckets also known as bins. In order to group the values in the bins or buckets binning method compare or consult with the neighborhood values and hence they perform local smoothing.

**2. Clustering**

In clustering some seed values are considered at the initial state and then the remaining data values from the setare then according placed into the cluster.

Each cluster will have specific set of values or data which is closest to the initially selected seed value for specific cluster.

The value which does not come under any cluster is the outlier or anomaly and it is considered to be noise, these outlier are then removed and data is smoothed.

**3. Combined computer and human inspection**

Many times combined effort of human and computer can be used for removing the outlier. In an application computer can be used to detect the pattern and then human can identify which to be considered and which to be not.

The pattern those are rejected can be considered as ther outlier and hence can be removed and the data can be smoothed.

**4. Regression**

In regression data values are plotted and then the fuction is evaluated which will fit maximum values on that fuction.

Such a function can be linear (Strainght line) or non linear (Curve) accordingly data is fitted on the shape defined by the fuction, the data values which are not covered by the fuction are considered to be outliers and hence they are removed as the unwanted/noisy values and hence the data can be smoothed.

**5.4.2. Aggregating**

Aggregation is similar to summarization operation on the data. Data cubes are generated by aggregation process.

For example in a retail shop daily sales data can be aggregated and the monthly and annual total sale amount can be calculated, which can further be used to identify profit and loss for the sales in that shop.

Data aggregation generally is used on large amount of data which does not give information as whole but after aggregation can give certain information.

Data aggregation is mainly used for gathering, utilization and presentation of the data which is available and can be presented on the internet.

Aggregation is generally done in order to get the summary and by using it the reports can be generated as and when needed. The tools or the functions used for aggregation should be well defined any wrong assumption can lead to wrong summary or report.

Aggregation can be done as and when needed but there are certain time intervals when the aggregation becomes compulsions.

* **Reporting period**

It is the time when the reports are to be generated and presented to the higher authorities in the business. Depending on the type of business the reporting period can be on daily, weekly, monthly, quarterly and yearly basis.

* **Granularity**

Granularity is a period over which data points for a particular resources or set of resources is collected. For example, if time series or real time data is used and you need to get the information about the particular share during 10:00 am to 10:05 am then this period of 5 minutes is said to be granule. Granularity can be in any range in minutes, in hours, in days, in weeks or even in months.

* **Polling period**

Polling period provides time duration which determines how often a particular data has been sampled. Polling period and granularity period can be same or different. Consider the granularity period is of 20 minutes and polling period is say 5 minutes then it can be concluded that in 20 minutes of the granularity period (20/5) 4 time sampling or aggregation is done.