# **Table of Contents**

[Table of Contents 1](#_Toc498183638)

[1. Data Validation 3](#_Toc498183639)

[1.1 Exploratory Data Analysis 3](#_Toc498183640)

[1.1.1 Variable Identification 3](#_Toc498183641)

[1.1.2 UNIvariant Analysis 3](#_Toc498183642)

[1.1.3 Bivariant Analysis 3](#_Toc498183643)

[1.2 Imputation of DATA 4](#_Toc498183644)

[1.2.1 Imputation - MICE 5](#_Toc498183645)

[1.2.2 Imputation - MissForest 6](#_Toc498183646)

[1.2.3 Spark EDA(Exploratory Data Analysis) 6](#_Toc498183647)

[1.3 DEmand Modeling 7](#_Toc498183648)

[1.3.1 Different Packages 7](#_Toc498183649)

[1.4 Scoring Model 7](#_Toc498183650)

# Data Validation

## Exploratory Data Analysis

https://www.analyticsvidhya.com/blog/2016/01/guide-data-exploration/

Remember the quality of your inputs decide the quality of your output. So, once you have got your business hypothesis ready, it makes sense to spend lot of time and efforts here. With my personal estimate, data exploration, cleaning and preparation can take up to 70% of your total project time. Below are the steps involved to understand, clean and prepare your data for building your predictive model:

1. Variable Identification
2. Univariate Analysis
3. Bi-variate Analysis
4. Missing values treatment
5. Outlier treatment
6. Variable transformation
7. Variable creation

Finally, we will need to iterate over steps 4 – 7 multiple times before we come up with our refined model.

### Variable Identification

First, identify Predictor (Input) and Target (output) variables. Next, identify the data type and category of the variables.

### UNIvariant Analysis

**Continuous Variable:**  We need to understand the central tendency (Mean,Median,Mode,Min,Max) and spread of the variable (Variance,Ranage,Quartile,IQR,Standard Deviation, Skewness & Kurtosis). Visualization through (Box Plot and Histogram).

**Categorical Variable:**  We’ll use frequency table to understand distribution of each category. Bar chart can be used as visualization.

### Bivariant Analysis

**Continuous & Continuous:** We should look at scatter plot. The pattern of scatter plot indicates the relationship between variables. The relationship can be linear or non-linear (correlation).

**Categorical & Categorical:** To find the relationship between two categorical variables, we can use following methods:

1. Two-way table:

We can start analyzing the relationship by creating a two-way table of count and count%. The rows represents the category of one variable and the columns represent the categories of the other variable. We show count or count% of observations available in each combination of row and column categories.

1. Stacked Column Chart:

This method is more of a visual form of Two-way table

**Chi-Square Test:** This test is used to derive the statistical significance of relationship between the variables. Also, it tests whether the evidence in the sample is strong enough to generalize that the relationship for a larger population as well. Chi-square is based on the difference between the expected and observed frequencies in one or more categories in the two-way table. It returns probability for the computed chi-square distribution with the degree of freedom.

* Probability of 0: It indicates that both categorical variable are dependent
* Probability of 1: It shows that both variables are independent

Probability less than 0.05: It indicates that the relationship between the variables is significant at 95% confidence.

**Categorical & Continuous:** While exploring relation between categorical and continuous variables, we can draw box plots for each level of categorical variables. If levels are small in number, it will not show the statistical significance. To look at the statistical significance we can perform Z-test, T-test or ANOVA.

## Imputation of DATA

* Data are missing completely at random (MCAR) when the probability of an instance (case) having a missing value for a variable does not depend on either the known values or the missing data.
* Data are missing at random (MAR) when the probability of an instance having a missing value for a variable may depend on the known values but not on the value of the missing data itself.
* Data are missing not at random (MNAR) when the probability of an instance having a missing value for a variable could depend on the value of that variable.

1. Mean,
2. K-nearest neighbors (KNN),
3. fuzzy K-means (FKM),
4. singular value decomposition (SVD),
5. bayesian principal component analysis (bPCA) and
6. multiple imputations by chained equations (MICE).

FKM is an extension of KNN based on fuzzy K-means clustering. SVD and bPCA are based on eigenvalues.

MICE are an iterative algorithm based on chained equations that uses an imputation model specified separately for each variable and involving the other variables as predictors.

Comparison was performed on four real datasets of various sizes, under a missing completely at random (MCAR) assumption, and based on four evaluation criteria:

1. Root mean squared error (RMSE),
2. unsupervised classification error (UCE),
3. supervised classification error (SCE) and
4. execution time.

Our results suggest that bPCA and FKM are two imputation methods of interest which deserve further consideration in practice

### Imputation - MICE

https://www.analyticsvidhya.com/blog/2016/03/tutorial-powerful-packages-imputing-missing-values/

MICE (Multivariate Imputation via Chained Equations) is one of the commonly used package by R users. Creating multiple imputations as compared to a single imputation (such as mean) takes care of uncertainty in missing values.

MICE assumes that the missing data are Missing at Random (MAR), which means that the probability that a value is missing depends only on observed value and can be predicted using them. It imputes data on a variable by variable basis by specifying an imputation model per variable.

Precisely, the methods used by this package are:

1. PMM (Predictive Mean Matching) – For numeric variables
2. logreg(Logistic Regression) – For Binary Variables( with 2 levels)
3. polyreg(Bayesian polytomous regression) – For Factor Variables (>= 2 levels)
4. Proportional odds model (ordered, >= 2 levels)

#load data

*> data <- iris*

#Get summary

*> summary(iris)*

Since, MICE assumes missing at random values. Let’s seed missing values in our data set using prodNA function. You can access this function by installing missForest package.

#Generate 10% missing values at Random

*> iris.mis <- prodNA(iris, noNA = 0.1)*

#Check missing values introduced in the data

*> summary(iris.mis)*

I’ve removed categorical variable. Let’s here focus on continuous values. To treat categorical variable, simply encode the levels and follow the procedure below.

#remove categorical variables

*> iris.mis <- subset(iris.mis, select = -c(Species))*

*> summary(iris.mis)*

#install MICE

*> install.packages("mice")*

*> library(mice)*

mice package has a function known as md.pattern(). It returns a tabular form of missing value present in each variable in a data set.

*> md.pattern(iris.mis)*

#For better visualization.

*> install.packages("VIM")*

*> library(VIM)*

*> mice\_plot <- aggr(iris.mis, col=c('navyblue','yellow'),*

*numbers=TRUE, sortVars=TRUE,*

*labels=names(iris.mis), cex.axis=.7,*

*gap=3, ylab=c("Missing data","Pattern"))*

### Imputation - MissForest

https://www.analyticsvidhya.com/blog/2016/03/tutorial-powerful-packages-imputing-missing-values/

missForest is an implementation of random forest algorithm. It’s a non parametric imputation method applicable to various variable types. Non-parametric method does not make explicit assumptions about functional form of f (any arbitary function). Instead, it tries to estimate f such that it can be as close to the data points without seeming impractical.

In simple words, it builds a random forest model for each variable. Then it uses the model to predict missing values in the variable with the help of observed values.

#missForest

*> install.packages("missForest")*

*> library(missForest)*

#load data

*> data("iris")*

#seed 10% missing values

*> iris.mis <- prodNA(iris, noNA = 0.1)*

*> summary(iris.mis)*

#impute missing values, using all parameters as default values

*> iris.imp <- missForest(iris.mis)*

#check imputed values

*> iris.imp$ximp*

#check imputation error

*> iris.imp$OOBerror*

### Spark EDA(Exploratory Data Analysis)

# to read the summary statistics of the dataframe

*df.describe().show()*

# to see for specific column

*df.describe(['column\_name']).show()*

#it shows count,mean,stddev,min,max

#to find the missing value find the total count

*df.count()*

#remove the rows with NA

*removeAllDF = df.na.drop()*

#Impute the mean value

*imputeDF = df*

*from pyspark.sql.functions import avg*

# to loop through columns in a dataframe

*for x in df.columns:*

*#meanValue = removeALLDF.agg(avg(x))*

#agg returns a dataframe, first row of the dataframe

*meanValue = removeALLDF.agg(avg(x)).first()[0]*

*print(x, meanValue)*

*imputeDF = imputeDF.na.fill(meanVlaue,[X])*

## DEmand Modeling

Demand modeling uses statistical methods and business intelligence inputs to generate accurate demand forecasts and effectively address demand variability.

“Normal distribution” assumption used by traditional models is totally inadequate, complicated by the increasing number of slow moving items, the so–called “long-tail”. In these scenarios, successfully managing forecasts and inventories requires advanced demand and inventory modeling technologies in order to reliably support high service levels.

### Different Packages

## Scoring Model

Scoring technology is typically applied to transactional data, sometimes in real time (credit card fraud detection, click fraud).

Scoring model is a special kind of predictive models. Process of applying a predictive model to a set of data is referred to as scoring the data. IBM® SPSS® Statistics has procedures for building predictive models such as regression, clustering, tree, and neural network models. ... You can then use that model file to generate predictive scores in other datasets.

Lead Score in Marketing:

Lead scoring is a methodology used to rank prospects against a scale that represents the perceived value each lead represents to the organization. The resulting score is used to determine which leads a receiving function (e.g. sales, partners, teleprospecting) will engage, in order of priority.

## Model Selection

### AIC (Akaike information criterion)

https://en.wikipedia.org/wiki/Akaike\_information\_criterion

The Akaike information criterion (AIC) is an estimator of the relative quality of statistical models for a given set of data. Given a collection of models for the data, AIC estimates the quality of each model, relative to each of the other models. Thus, AIC provides a means for model selection.

AIC is founded on information theory: it offers an estimate of the relative information lost when a given model is used to represent the process that generated the data. (In doing so, it deals with the trade-off between the goodness of fit of the model and the complexity of the model.)

AIC does not provide a test of a model in the sense of testing a null hypothesis. It tells nothing about the absolute quality of a model, only the quality relative to other models. Thus, if all the candidate models fit poorly, AIC will not give any warning of that.

As an example, suppose that there are three candidate models, whose AIC values are 100, 102, and 110. Then the second model is exp((100 − 102)/2) = 0.368 times as probable as the first model to minimize the information loss. Similarly, the third model is exp((100 − 110)/2) = 0.007 times as probable as the first model to minimize the information loss.

In this example, we would omit the third model from further consideration. We then have three options: (1) gather more data, in the hope that this will allow clearly distinguishing between the first two models; (2) simply conclude that the data is insufficient to support selecting one model from among the first two; (3) take a weighted average of the first two models, with weights proportional to 1 and 0.368, respectively, and then do statistical inference based on the weighted multimodel.[4]

### BIC - Bayesian information criterion

When fitting models, it is possible to increase the likelihood by adding parameters, but doing so may result in overfitting. Both BIC and AIC attempt to resolve this problem by introducing a penalty term for the number of parameters in the model; the penalty term is larger in BIC than in AIC.

The BIC criterion suffers from two main limitations:

1. the above approximation is only valid for sample size { n} much larger than the number {k} of parameters in the model.

2. BIC cannot handle complex collections of models as in the variable selection (or feature selection) problem in high-dimension.[3]