PREDICTION PERFORMANCE OF VERY LOW BIRTH WEIGHT INFANT

PROJECT REPORT SUBMITTED TOMANGALORE UNIVERSITY

IN PARTIAL FULLFILMENTS OF THE REQUIREMENTSFOR THE COMPLETION OF MASTERS DEGREE IN STATISTICS

SUBMITTED BY

NITHIN DS

M.Sc. IV SEMESTER

UNDER THE GUIDANCE OF

Ms. PREETHI J SHETTY

DEPARTMENT OF PG STUDIES ANDRESEARCH IN STATISTICS

MANGALORE UNIVERSITY MANGALAGANGOTHRI - 574199

SEPTEMBER - 2022

MANGALORE UNIVERSITY

DEPARTMENT OF PG STUDIES AND RESEARCH IN STATISTICS

MANGALAGANGOTHRI

CERTIFICATE

Certified that this is the bonafide record of the project work done by **Mr. NITHIN DS** during the year 2021- 22 as a part of her M.Sc. (Statistics) fourth semester course.

Reg. No:

2	**0**	**1**	**6**	**9**	**1**	**4**	**0**	**6**	**1**	**1**	**9**

Chairman of the department

(Ms.Preethi Jayaram Shetty) **(Prof.Ishwara P)**

Place: Mangalagangothri

Date:

ACKNOWLEDGEMENT

First of all, I would like to thank God Almighty for showering his gracious blessings on throughout the project.

I express my profound gratitude to Prof. Ishwar P, Chairman, Departmentof PG Studies and Research in Statistics, Mangalore University, whole-heartedly for his motivations, support and inspiration.

I am deeply grateful to Ms. Preethi Jayarama Shetty, Department of PG Studies and Research in Statistics, Mangalore University, for her continuous help, support and guidance throughout this project.

I would like to express my gratitude to the faculty members, Mr. Satyanarayana, Ms. Prajna R, Dr. Harsha S, Ms. Preethi Jayaram Shetty , Ms. Sukshitha R and

Ms. Poornima for their constant encouragement.

I thank all my classmates and friends who have helped me directly or indirectlyto complete the project successfully. My special thanks to my parents whose encouragement and support has helped me to rise to this level.

Date:

Place: Mangalagangothri (NITHIN DS)

DECLARATION

I, NITHIN DS hereby declare that the matter embodied in thisreport entitled **"PREDICTION PERFORMANCE OF VERY LOW BIRTH WEIGHT INFANT"** is a bonafide record of project work carried out by me under the guidance and supervision of

Ms. Preethi Jayarama Shetty,Department PG studies and Research in of ,Statistics, MangaloreUniversity. I further declare that no part of the work contained in the report has previously been formed the basis for the award of any Degree, Diploma, Fellowship or any other similar title or recognitionof any other University.

Date:

Place: Mangalagangothri (NITHIN DS)

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Introduction to Low Birth Weight:

Low birth weight is defined as infant born with weight of less than 2500 g. It is one of the major public health problems worldwide. Low birth weight contributes to 60 to 80 percent of neonatal deaths worldwide. Approximately 20 million low birth weight babies are born every year, most of them in developing countries. Preterm births, those occurring before the 37th week of pregnancy, are also on the rise. They account for over 10 percent of births globally. Most premature infants also have low birth weights. Complications from prematurity and low birth weight are the leading cause of death for children under age five around the world. Many survivors of preterm birth and low birth weight may suffer from disabilities such as learning problems, hearing impairment and vision issues.

The normal range of birth weight is from 5.5 to 8.75 pounds. Delivery after the 37th week of pregnancy is considered "at term." Any baby born weighing less than 5.5 pounds is low birth weight, and a baby born before 37 weeks is premature.

About one in 12 or eight percent of babies in the United States is born underweight, and about two-thirds of these low birth weight babies are also premature. A small percentage, about 1.4 percent of babies have very low birth weight – under 3.5 pounds. Almost all of these babies are premature, and some are extremely premature – born at 25 weeks or earlier.

The number of babies born with low birth weight is increasing along with the number of multiple births in the United States. More than half of multiple birth babies have low birth weight, and six percent of single birth babies are born with low birth weight.

Causes And Risk factors:

The main cause of low birth weight is premature birth, or any delivery before 37 weeks. The baby has not had enough time in the uterus to grow and gain weight. Babies born at term who are underweight may still be small and weak. Premature babies will be both small and not yet fully developed.

Intrauterine Growth Restriction (IUGR) and SGA are not the same thing, although they often appear together. IUGR is when the baby's weight is under the 10th percentile for their gestational age before they are born. It is diagnosed by ultrasound. IUGR indicates the fetus has not reached its full potential for growth because of some environmental factor in the womb or some genetic factor. SGA refers to a baby who weighs less than the 10th percentile for their gestational age at birth.

With IUGR, the baby may be developing too slowly due to infections in the uterus or due to birth defects. Babies with birth defects are more likely to be born prematurely and also with low birth weights. The doctor will want to establish an accurate gestational date for the pregnancy, starting the count from the first day of the last menstrual cycle. Once the gestational age of the fetus is known, the doctor can more easily determine whether the fundal height coincides with the expected size of the baby. The doctor can also compare ultrasound findings with expectations for a fetus of that age.

Some women are more at risk for having babies with IUGR. Risk factors in the mother include:

- Weight of the mother under 100 pounds.
- Poor prenatal nutrition
- Use of alcohol, cigarettes or drugs
- Pregnancy-induced high blood pressure or preeclampsia
- Gestational diabetes
- Chronic health conditions like diabetes, heart problems, lung or kidney problems, and high blood pressure
- Taking medications for conditions including blood clots, seizures and high blood pressure
- Inadequate weight gain during pregnancy
- Birth of a previous low birth weight baby.
- Age under 17 or over 35
- Racial and ethnic factors: Women of colour in the United States are between seven and 13 times more likely to have a baby with low birth weight.
- Socioeconomic factors, such as poverty, low education and exposure to domestic violence.

Conditions inside the uterus may also cause the fetus difficulty in getting the nutrition it needs to grow and develop. These include:

- Birth defects and chromosomal abnormalities
- Problems or abnormalities with the placenta
- Problems with the umbilical cord.
- Being a fetus with a twin or triplet (or more) in the uterus.
- Inadequate amniotic fluid (oligohydramnios)
- Infections in the uterus, such as rubella, chicken pox, sexually transmitted infections, cytomegalovirus or toxoplasmosis
- Birth by preterm labor shortening the time available to grow and develop.

Some Complications for Babies with Low Birth Weight:

Premature babies born around 32 weeks may struggle with breathing, staying warm and eating. Babies born very early, after fewer than six months or 26 weeks gestation, are the ones most likely to have more serious problems.

Premature babies may have problems breathing, called respiratory distress syndrome (RDS). Babies with RDS lack a protein called surfactant, which helps keep small air sacs in their lungs from collapsing. They receive this surfactant in the neonatal intensive care unit (NICU) to help their lungs work better and mature. They may also get supplemental oxygen.

Some premature babies and babies with low birth weight develop intraventricular hemorrhage, or brain bleeds. Most are mild and fix themselves, and leave no ongoing problems. More severe bleeds cause fluid buildup and pressure in the brain, which can cause brain damage. The baby may require surgery and the insertion of a tube to drain the excess fluid and prevent damage.

One problem seen more often is patent ductus arteriosus, in which an opening between two of the major blood vessels leading away from the heart has not yet closed properly. This leads to extra blood flow into the lungs. In some cases, the hole closes on its own after a few days. In other babies, surgery or medication may be required.

Tiny babies also have immature intestines that are more prone to necrotizing enterocolitis. This leads to swelling of the belly, feeding problems, and possibly surgery to remove damaged parts of the intestine.

Many premature infants are born with the retinas of their eyes not yet fully developed. Retinopathy of prematurity affects both eyes and requires immediate treatment for the baby to keep their vision.

Low birth weight and premature infants may have livers that are not working well or are not fully developed. They cannot process the chemical bilirubin out of their body and develop jaundice. Their eyes and skin turn yellow because of the excess bilirubin in their blood. Treatment includes time under special lights to transform the bilirubin into a chemical that is easier to rid from the body, IV treatment with immunoglobulin blood proteins, or even blood transfusions.

Because many low birth weight babies are also premature, doctors may have trouble telling which problems are due to the prematurity and which are due to the low birth weight. Regardless, the smaller a baby is at birth, the more likely he or she will have complications. Their survival depends on how well they gain weight and avoid infections.

Babies born with IUGR are statistically at higher risk for:

- Birth by cesarean section.
- Lack of oxygen (hypoxia) at birth.
- Meconium aspiration, where the baby swallows part of its first bowel movement. The alveoli sacs in the baby's lungs become distended and a pneumothorax, or collapsed lung, can occur. The baby may develop bacterial pneumonia.
- Hypoglycemia, or low blood sugar, at birth.
- Polycythemia, an increased number of red blood cells, related to lack of oxygen.
- Decreased blood flow (hyperviscosity) due to excess red blood cells.
- Increased risk of neurological and motor disabilities and delays.
- Higher risk of social delays and learning disabilities.

Treatment:

Treatment for the baby is determined by the doctor depending on the baby's gestational age, health and ability to tolerate treatments. Many times, care of a low birth weight or preterm baby includes spending time in the NICU, a temperature-controlled incubator, and special feeding through a stomach tube or IV line. Typically, low birthweight babies can catch up in growth if they have no other complications Low birth weight babies need to receive enough nutrition to grow in the NICU at about the rate they would have if they had stayed in the uterus. For a tiny baby born at 24 weeks, this may mean gaining about five grams (0.20 ounce) per day at first. A larger baby born around 33 weeks may need to gain 20-30 grams (0.7 to 1.1 oz) per day. Generally, a low birth weight baby should gain about ¼ ounce per day for every pound they weigh. This is the average rate a fetus grows throughout the third trimester of pregnancy.

Very premature babies may have trouble sucking from a breast or bottle because they cannot coordinate sucking, swallowing and breathing. Less premature babies often do best breastfeeding. A bottle nipple may cause them problems because it is harder to control.

Even the smallest low birth weight and premature babies benefit from breast milk, whether it comes from breastfeeding, a bottle, or via a tube feeding. Breast milk contains antibodies from the mother that help protect the baby from infections, Sudden Infant Death Syndrome (SIDS), and necrotizing enterocolitis (NEC). If the mother cannot produce enough milk, many NICUs

will give babies breast milk from a milk bank because breast milk has benefits that cannot be replicated with formula.

After birth, it is important to guard low birth weight babies against infections, breathing problems and dehydration. These babies lose more water than babies born at term or normal weights because their kidneys are not developed enough to control water levels. The nurses in the NICU keep close watch on the baby's fluid intake and output to ensure they stay properly hydrated.

1.2 Objectives:

- To identify the risk factors causing the LBW.
- Classification using statistical models and various machine learning techniques.
- Selecting overall best classifier by comparing the performance of classification techniques based on Accuracy, sensitivity and specificity.
- Predicting whether the infant is likely to get survive or not

1.3 About the data:

To meet the above objectives a secondary data of very low birth weight infant is collected from the website https://hbiostat.org/data/ .The data includes 382records and 24 attributes The variables description are given below.

- Birth = Birth is continuous variable. Date of birth (admission)
- Exit =Exit Date of death or discharge
- Hospital Stay= Hospital stays in number of days
- Lowph= Lowest pH in first 4 days of life
- Pltct= Platelet count (*10^9/L)
- Race(Black=0; White=1; Native india=2; Oriental=3
- Bwt=Birth weight in gram
- Gest=Gestational age in weeks
- Twn=Multiple gestation, No=0, Yes=1

- Lol=Duration of labor in hours
- Magsulf=Mother treated with MgSO4. No=0, Yes=1
- Meth=Mother treated with beta-methasone. No=0, Yes=1
- Toc=Tocolysis - mother treated with beta-adrenergic drug. No=0, Yes=1
- Delivery=Abdominal and Vaginal; Abdominal=0, Vaginal=1
- Apg1=Apgar (The Apgar score, the very first test given to a new-born) at one minute
- Vent=Assisted ventilation used. No=0, Yes=1
- Pda=Patent ductus arteriosus detected No=0, Yes=1
- Pneumo=Pneumothorax occurred No=0, Yes=1
- Cld=On suppl. oxygen at 30 days No=0, Yes=1
- Pvh=Periventricular haemorrhage Absent=0, Definite=1, Possible=2
- Ivh=Intraventricular haemorrhage Absent=0, Definite=1, Possible=2
- Sex=Gender, Male= 0, Female=1
- Dead=Live status, Live=0, Dead=1

1.4 Software used

R software

For the data analysis and representation of the data, we have used R software of version 4.2.1. We selected R as it is one of the widely used public domain software for data analysis.

1.5 Summary of the project

In chapter 3 of this project, the different techniques like preliminary analysis which includes descriptive statistics and various machine leaning techniques were summarized. In chapte4, the various results found throughout analysis were arranged systematically with necessary references. The major findings of the study and the conclusions were given in the last chapter of this project.

Chapter 2
LITERATURE REVIEW

Ashok Kumar Dwivedi *et al,* (2016) recommended different algorithms like Naive Bayes, Classification Tree, KNN, Logistic Regression, SVM and ANN. The Logistic Regression gives better accuracy compared to other algorithms.

Nitesh V. Chawla *et al,* (2016)in their paper "Synthetic Minority Over-sampling Technique" This paper shows that a combination of our method of over-sampling the minority (abnormal) class and under-sampling the majority (normal) class can achieve better classifier performance (in ROC space) than only under-sampling the majority class. This paper also shows that a combination of our method of over-sampling the minority class and under-sampling the majority class can achieve better classifier performance (in ROC space) than varying the loss ratios in Ripper or class priors in Naive Bayes

Nicola Lunardon *et al,* (2014)In their paper" ROSE: A Package for Binary Imbalanced Learning" The ROSE package provides functions to deal with binary classification problems in the presence of imbalanced classes. Functions that implement more traditional remedies for the class imbalance and different metrics to evaluate accuracy are also provided. These are estimated by holdout, bootstrap, or cross-validation methods.

Namik Y. Ozbek *et al,* (2019)In their paper" Assessment of Low Birth Weight and Associated Factors Among Neonates in Butajira General Hospital, South Ethiopia, Cross Sectional Study" The main objective of this study was to assess the prevalence and associated factors of low birth weight among newborns delivered at Butajira General Hospital, Southwest Ethiopia.

Chapter 3
METHEDOLOGY

3.1 Descriptive Statistics

Descriptive statistics are brief descriptive coefficients that summarize a given data set, which can be either a representation of the entire population or a sample of a population. Descriptive statistics are broken down into measures of central tendency and measures of variability (spread). Measures of central tendency include the mean, median, and mode, while measures of variability include standard deviation, variance, minimum and maximum variables, kurtosis, and Skewness . All descriptive statistics are either measures of central tendency or measures of variability, also known as measures of dispersion.

Central Tendency

Measures of central tendency focus on the average or middle values of data sets, whereas measures of variability focus on the dispersion of data. These two measures use graphs, tables and general discussions to help people understand the meaning of the analysed data.

Measures of central tendency describe the centre position of a distribution for a data set. A person analyses the frequency of each data point in the distribution and describes it using the mean, median, or mode, which measures the most common patterns of the analysed data set.

Mean

The Arithmetic mean is commonly known as average. The average of a given set of numbers is called the arithmetic mean, or simply, the mean of the given numbers. Thus, the arithmetic mean of a group of observations is defined as

$$\bar{X} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

$\bar{x}$ = the mean of xi

xi= each of the values of the data.

n =number of data points

Mode :

Mode is a statistical term that refers to the most frequently occurring number found in a set of numbers. The mode is found by collecting and organizing data in order to count the frequency of each result. The result with the highest number of occurrences is the mode of the set.

Measures of Variability

Measures of variability (or the measures of spread) aid in analysing how dispersed the distribution is for a set of data. For example, while the measures of central tendency may give a person the average of a data set, it does not describe how the data is distributed within the set.

Range, quartiles, absolute deviation, and variance are all examples of measures of variability.

Standard Deviation

Standard deviation is a measure of the dispersion of a set of data from its mean. It is calculated as the square root of variance by determining the variation between each data point relative to the mean. If the data points are further from the mean, there is higher deviation within the data set.

$$SD = \sqrt{\frac{\sum_{i=1}^{n}(xi - \overline{x})^2}{(n-1)}}$$

$\bar{x}$ = the mean of xi

xi= Each of the values of the data.

n =number of data points

MEASURE OF SHAPE KURTOSIS

In probability theory and statistics, kurtosis is any measure of the peakedness" of the probability distribution of a real- valued random variable. In a similar way to the concept of skewness, kurtosis is a descriptor of the shape of a probability distribution and just as for skewness, there are different ways of quantifying it for a theoretical distribution and corresponding ways of estimating it from a sample from a population. One common measure of kurtosis, originating with Karl Pearson, is based on a scaled version of the fourth moment of the data or population.

The fourth standardized moment is defined as,

$$B^2 = \frac{E[(X-\mu)^4]}{(E[(X-\mu)^2])^2} = \frac{\mu_4}{\sigma_4}$$

Where μ4 is the fourth moment about the mean and σ is the standard deviation. Distributions with zero excess kurtosis are called mesokurtic. A distribution with positive excess kurtosis is called leptokurtic. A distribution with negative excess kurtosis is called platykurtic.

Types of Kurtosis

There are three categories of kurtosis that can be displayed by a set of data. All measures of kurtosis are compared against a standard normal distribution, or bell curve.

The first category of kurtosis is a mesokurtic distribution. This type of kurtosis is the most similar to a standard normal distribution in that it also resembles a bell curve. However, a graph that is mesokurtic has fatter tails than a standard normal distribution and has a slightly lower peak. This type of kurtosis is considered normally distributed but is not a standard normal distribution.

The second category is a leptokurtic distribution. Any distribution that is leptokurtic displays greater kurtosis than a mesokurtic distribution. Characteristics of this type of distribution is one with extremely thick tails and a very thin and tall peak. The prefix of "lepto-" means "skinny," making the shape of a leptokurtic distribution easier to remember. T-distributions are leptokurtic.

The final type of distribution is a platykurtic distribution. These type of distributions have slender tails and a peak that's smaller than a mesokurtic distribution. The prefix of "platy-" means "broad," and it is meant to describe a short and broad-looking peak. Uniform distributions are platykurtic.

SKEWNESS

In probability theory and statistics, skewness is a measure of the extent to which a probability distribution of a real -valued random variable "leans" to one side of the mean. The skewness value can be positive or negative or even undefined. For unimodal distribution, negative skew indicates that the tail on the left side of the probability density function is longer or fatter than the right side- it does not distinguish these shapes. Conversely, positive skew indicates that the tail on the right side is longer or fatter than the left side. In cases where one tail is long but the other tail is fat, skewness does not obey a simple rule. The skewness of a random variable X is the third standardized moment, denoted and defined as,

$$\gamma_1 = \mathrm{E}[(\frac{X-\mu}{\sigma})^3] = \frac{\mu_3}{\sigma_3} = \frac{E[(X-\mu)^3]}{E[(X-\mu)^2]^{3/2}} = \frac{k_3}{k_2^{3/2}}$$

Where μ3 is the third moment about the mean μ,σ is the standard deviation , and E is the expectation operator. The last equality expresses skewness in terms of the ratio of the third cumulate k3 and the 1.5th power of the second cumulate k2. This analogous to the definition of kurtosis as the fourth cumulate normalized by the square of the second cumulate.

3.2 Dummy Variables

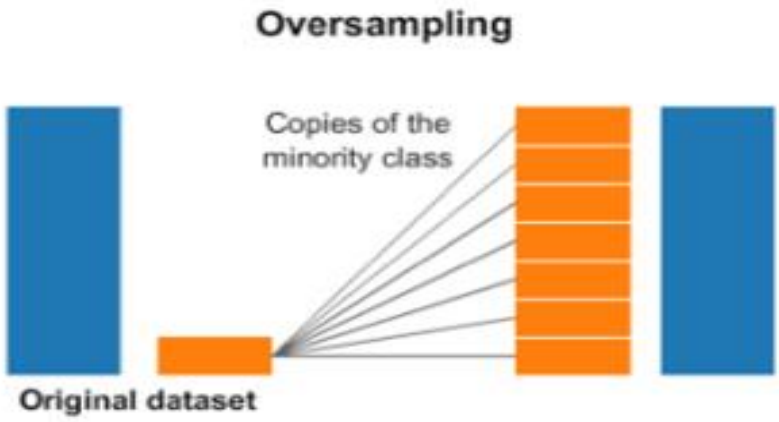
A dummy variable is a variable that takes values of 0 and 1, where the values indicate the presence or absence of something. Where a categorical variable has more than two categories, it can be represented by a set of dummy variables**,** with one variable for each category. Numeric variables can also be dummy coded to explore nonlinear effects**.** Dummy variables are also known as indicator variables**,** design variables**,** contrasts**,** one-hot coding, and binary basis variables. Dummy variables assign the numbers '0' and '1' to indicate membership in any mutually exclusive and exhaustive category. The number of dummy variables necessary to represent a single attribute variable is equal to the number of levels (categories) in that variable minus one. For a given attribute variable, none of the dummy variables constructed

can be redundant. That is, one dummy variable cannot be a constant multiple or a simple linear relation of another.

3.3 Sampling techniques to handle Imbalanced Data

- **Synthetic Minority Oversampling Technique:**

SMOTE (synthetic minority oversampling technique). It aims to balance class distribution by randomly increasing minority class examples by replicating them. It synthesizes new minority instances between existing minority instances. It generates the virtual training records by linear interpolation for the minority class. These synthetic training records are generated by randomly selecting one or more of the k-nearest neighbors for each example in the minority class. After the oversampling process, the data is reconstructed and several classification models can be applied for the processed data



- **Random Over-Sampling technique:**

ROSE (Random Over-Sampling) is a bootstrap-based technique which aids the task of binary classification in the presence of rare classes. It handles both continuous and categorical data by generating synthetic examples from a conditional density estimate of the two classes. It uses smoothed

bootstrapping to draw artificial samples from the feature space neighbourhood around the minority class. SMOTE draws artificial samples by choosing points that lie on the line connecting the rare observation to one of its nearest neighbours in the feature space.

3.4 Logistic Regression

Logistic regression analysis studies the association between a categorical dependent variable and a set of independent (explanatory) variables, and estimates the probability of occurrence of an event by fitting data to logistic curve. There are two models of logistic regression, binary logistic regression and multinomial logistic regression. Binary logistic regression is typically used when the dependent variable is dichotomous and the independent variables are either continuous or categorical. When the dependent variable is not dichotomous and is comprised of more than two categories, a multinomial logistic regression can be employed. Logistic regression competes with discriminant analysis as a method for analysing categorical response variables.

Suppose the numerical values of 0 and 1 are assigned to the two outcomes of the binary variables that are called dummy variable.

A Dummy variable is variable that takes values of 0 and 1, where the values indicate the presence or absence of categorical effect. Where a categorical variable has more than two categories, it can be represented by a set of dummy variables, with one variable for each category. Numeric variable can also be dummy coded to explore the nonlinear effects. Dummy variables are also known as indicator variables, design variables, contrasts, one-hot coding and binary basis variables.

Dummy variables assign the numbers '0' and '1' two indicate the membership in any mutually exclusive and exhaustive category. The number of dummy variables necessary to represent a single attribute variable is equal to the number of levels(categories) in that variable minus one. For a given attribute variable, none of the dummy variables constructed can be redundant. That is, one dummy variable cannot be a constant multiple or a single linear relation of another.

Often, 0 represents a negative response and the 1 represents the positive response. The mean of this variable will be the proportion of positive responses. If 'p' is the probability of observations with an outcome of 1, then '1- p' is the probability of an outcome 0. The ratio $\frac{p}{1-p}$ is called the odds and the logit is the logarithm of the odds or just log odds. Mathematically, the logit transformation is written as

$$1=\text{logit}(p) = \ln\left(\frac{p}{1-p}\right)$$

p ranges between 0 and 1. The logit ranges from minus infinity to plus infinity, note that the zero logit occurs when p is 0.5. The logistic transformation is the inverse of the logit transformation. It is written as,

$$p = logistic(l) = \frac{el}{1-el}$$

The Log Odds Ratio Transformation: The difference between two log odds can be used to compare two proportions, such as that of YES versus NO. Mathematically, this difference is written as follows:

$$l1 - l2 = logit\,(p1) - logit\,(p2)$$

$$=\ln\left(\frac{p1}{1-p1}\right) - \ln\left(\frac{p2}{1-p2}\right)$$

$$= ln\left(\frac{\frac{p_1}{1-p_1}}{\frac{p_2}{1-p_2}}\right)$$

$$= \ln\frac{(p_1(1-p_2)}{(p_2(1-p_1)}$$

This difference is often referred to as the log odds ratio. The odds ratio is often used to compare proportions across groups. Note that the logistic transformation is closely related to the odds ratio. The reverse relationship is given by,

$$\frac{p_{1(1-p_2)}}{p_{2(1-p_1)}} = e\,(l1-l2)$$

The logistic Regression and Logit Models: In logistic regression, a categorical dependent variable Y having G (Usually G=2) unique values is regressed on a set of p independent variables x_1, x_2, …, xn. For example, Y may be presence or absence of a disease, condition after surgery, or marital status. Since the names of these partitions are arbitrary, we often refer to them by consecutive numbers. Let X=(X_1, X_2 , … , Xp)

$$\text{Bg} = (, \dots , \beta gp)'$$

The logistic regression model is given by the G equations

$$\ln(pg/p_1) = \ln(pg/p_1) + \beta g_1 X_1 + \cdots + \beta gpXp = \ln(pg/p_1) + XBg$$

here, pg is the probability that an individual with values X_1, X_2, …, Xp is in outcome g. That is,

$$pg = P(Y = g|X)$$

Usually $X1 \equiv 1$ (i.e., an intercept is included), but this is not necessary.

The quantities p_1, $p2$, …, pG represent the prior probabilities of outcome membership. If these prior probabilities are assumed equal, then the term $ln\,(pg/\,p1)$ becomes zero and drops

out. If the priors are not assumed equal, they change the values of the intercepts in the logistic regression equation.

Outcome one is called the reference value. The regression coefficients β_{11}, β_{12}, …, β_{1p} for the reference value are set to zero. The choice of the reference value is arbitrary. Usually, it is the most frequent value or a control outcome to which the other outcomes are to be compared. This leaves G-1 logistic regression equations in the logistic model. Theβ's are population regression coefficients that are to be estimated from the data. Their estimates are represented by b's. The β's represents unknown parameters to be estimated, while the b's are their estimates.

These equations are linear in the logits of p. However, in terms of the probabilities, they are nonlinear. The corresponding nonlinear equations are

$$pg = P(Y = g|X) = \frac{e^{XBg}}{1+eXB2+eXB3+\cdots+eXBG}$$

Here,e^{XB1} = 1 because all of its regression coefficients are zero.

Assumptions of Logistic Regression

Logistic regression does not require many of the principal assumptions of linear regression models that are based on ordinary least square method- particularly regarding linearly of relationship between the dependent and independent variables, normality of the error distribution, homoscedasticity of the errors, and measurement level of the independent variables. Logistic regression can handle non-linear relationship between the dependent and independent variables, because it applies a non-linear log transformation of the linear regression. The error terms (the residuals) do not need to be multivariate normally distributed-although multivariate normality yields a more stable solution. The variance of errors can be heteroscedastic for each level of the independent variables. Logistic regression can handle not only continuous data but also discrete data as independent variable.

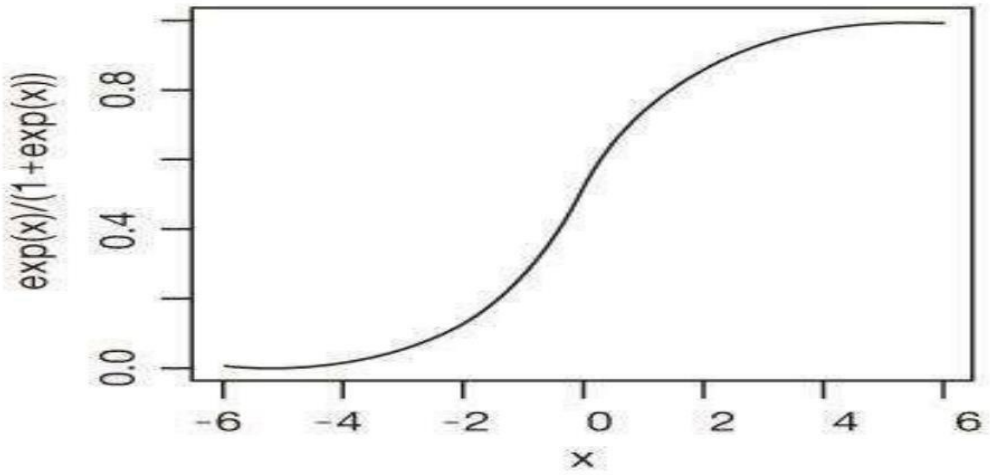


Figure 1: Graph of logistic curve where α=0 and β=1.

Fitting the Logistic Regression Model:

Although logistic regression model, $logit(y) = \alpha + \beta X$ looks similar to a simple linear regression model, the underlying distribution is binomial and the parameters, α and β cannot be estimated in the same way as for simple linear regression. Instead, the parameters are usually estimated using the method of maximum likelihood of observing the sample values. Maximum likelihood will provide values of α and β which maximize the probability of obtaining the dataset. It requires iterative computing with computer software. The likelihood 24 function is used to estimate the probability of observing the data, given the unknown parameters (α and β). A "Likelihood" is a probability that the observed values of the dependent variable may be predicted from the observed values of the independent variables. The likelihood varies from 0 and 1 like any other probabilities. Practically, it is easier to work with the logarithm of the likelihood function. This function is known as the log-likelihood. Log-likelihood will be used for inference testing when comparing several models.

The log likelihood 19 varies from 0 to -∞ (it is negative because the natural log of any number less than 1 is negative). In logistic regression, we observe binary outcome and predictors, and we wish to draw inferences about the probability of an event in the population. Suppose in a population from which we are sampling, each individual has the same probability p that an event occurs. For each individual in our sample of size n, $Yi = 1$ indicates that an event occurs for the ith subject, otherwise, $Y_i=0$ The observed data are $Y_1, Y_2, \cdots, Y_n$ and $X_1, X_2, \cdots, X_n$. The joint probability of the data is given by,

$$L= \prod_{i=1}^{n} \pi(x)^{yi} \ (1 - \pi(x))^{\ 1-yi}$$

Natural logarithm of the likelihood is

$$l=\log (L) = \sum Yi \ n \ i=1 \log ((x)) + (n - \sum Yi \ n \ i=1) \log (1 - \pi(x))$$

In which

$$(x) = p \ (y/x) = \frac{e^{\ \alpha+\beta x}}{1+e^{\ \alpha+\beta x}}$$

Estimating the parameters α and β is done using the first derivatives of log likelihood. And solving them for α and β. For this iterative computing is used. An arbitrary value for the coefficients (usually 0) is first chosen. Then log-likelihood is compared and variation of coefficients values observed. Reiteration is then performed until maximization of l (equivalent to maximizing L). The results are the maximum likelihood estimators of α and β.

Stepwise Logistic Regression:

Stepwise Logistic Regression is most often used in situations where the "important" independent variables are not known and association with the outcome not well understood. In these instances, most studies will collect many possible independent variables and screen them for significance. Stepwise logistic regression offers a fast and effective means of screening a large number of variables, and simultaneously fit a number of logistic regression equations. There are two basic forms of stepwise logistic regression: forward inclusion and backward elimination. In forward logistic regression all independent variables are initially withheld from the model. At subsequent steps in the procedure, those variables determined to be significant are added to the model while all others are withheld. Just the opposite occurs in backward logistic regression in which all independent variables are initially include in the model. At subsequent steps in the procedure, those variables determined insignificant are eliminated for the model until the remaining variables are all deemed "important". In stepwise logistic regression, selection or deletion of variables from the model is based on a statistical algorithm that checks for "importance" of variables, and either includes or excludes them on the basis of a fixed decision rule. The likelihood ratio Chi-square test used to assess significance in logistic regression since the errors are assumed to follow a binomial distribution.

This test assigns a p-value to each variable to assess significance. Therefore, the most important variable is the one with the smallest p-value. An important element of stepwise logistic regression is selection of removal and entry criteria to determine variable significance. The removal criterion is the p-value used to eliminate insignificant independent variables. If a variable's p-value is equal to or greater than number it will be eliminated from the model. The entry criterion 26 value determines which independent variables will be included in the model. If a variable's p-value is less than this value then it will be entered into the model.

Predictive Accuracy and Discrimination

The classification table is a method to evaluate the predictive accuracy of the logistic regression model. In this table observed values for the dependent outcomes and the predicted values (at a user defined cut-off) are cross-classified. For example, if a cut-off value is 0.5, all predicted values above 0.5 can be classified as predicting an event, and all below 0.5 as not predicting the event. Then a two-by-two table of data can be constructed with dichotomous observed outcomes, and dichotomous predicted outcomes.

Classification Table (Confusion Matrix): A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. The confusion matrix itself is relatively simple to understand, but the related terminology can be confusing.

Receiver operating characteristic (ROC) curve

ROC curve summarizes the model's performance by evaluating the trade-offs between true positive rate (sensitivity) and false positive rate (1- specificity). ROC curve is simply a plot of the values of sensitivity against one minus specificity, as the value of the cut-point cc is increased from 0 through to 1. A model with high discrimination ability will have high sensitivity and specificity simultaneously, leading to an ROC curve which goes close to the top left corner of the plot. A model with no discrimination ability will have an ROC curve which is the 45 degree diagonal line.

The area under curve (AUC), referred to as index of accuracy, is a perfect performance metric for ROC curve. Higher the area under curve, better the prediction power of the model.

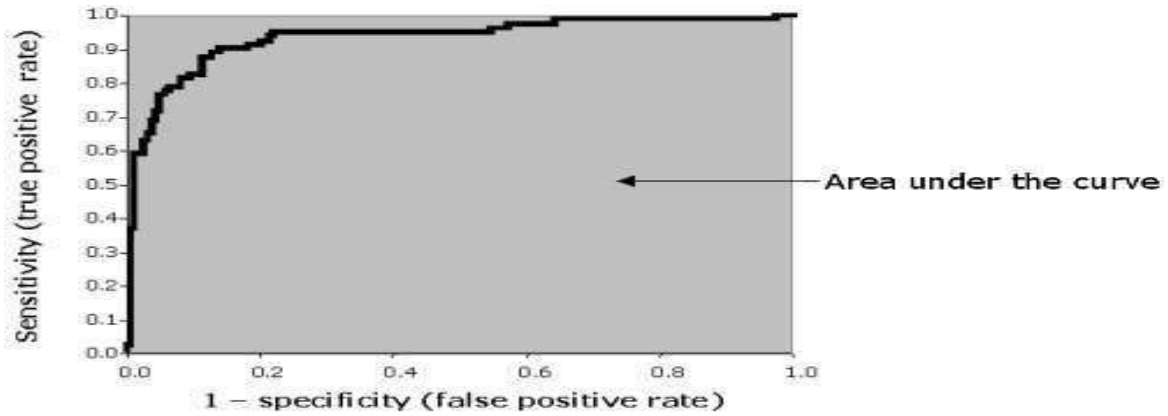


Figure 2

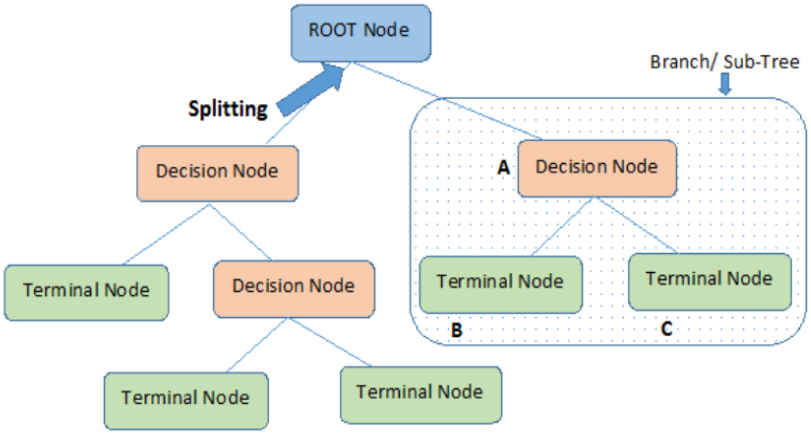
The accuracy of the test depends on how well the test separates the group being tested into those with and without the disease in question. Accuracy is measured by the area under the ROC curve. An area of 1 represents a perfect test; an area of 0.50 represents a worthless test. A rough guide for classifying the accuracy of a diagnostic test is the traditional academic point system:

- 0.90-1 = excellent (A)
- 0.80-0.90 = good (B)

- 0.70-0.80 = fair (C)
- 0.60-0.70 = poor (D)
- 0.50-0.60 = fail (F)

3.5 Decision Tree

A Decision tree is a flowchart-like tree structure, where each internal node (non-leaf node) denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (or *terminal node*) holds a class label. The topmost node in a tree is the root node.



- Given a tuple ***X***, for which the associated class label is unknown, the attribute values of the tuple are tested against the decision tree.
- A path is traced from the root to a leaf node, which holds the class prediction/predicted values for that tuple.

The construction of decision tree does not require any domain knowledge or parameter setting, and therefore is appropriate for exploratory knowledge discovery. Decision trees can handle multidimensional data. Their representation of acquired knowledge in tree form is intuitive and generally easy to assimilate by humans. In general, decision tree have good accuracy. However, successful use may depend on the data at hand. The average of values dependent variable of the tuple is taken as the predicted value for all those tuples.

Splitting scenarios

There are three possible scenarios, Let A be the splitting attribute. A has v distinct values, a_1, a_2,…, a_v , based on the training data.

A is discrete-valued. In this case, the outcomes of the test at node N correspond directly to the known values of A. A branch is created for each known value, a_j, of A and labelled with that value. Partition D_j is the subset of class-labelled tuples in D having value a_j of A.

A is continuous-valued: In this case, the test at node N has two possible outcomes, corresponding to the conditions A split point and A>split point,-respectively, where split point is the split-point returned by Attribute selection method as part of the splitting criterion. (In practice, the split point, a, is often taken as the midpoint of two known adjacent values of A and therefore may not actually be a pre-existing value of A from the training data.) Two branches are grown from N and labelled according to the previous outcomes. The tuples are partitioned such that D_1 holds the subset of class-labelled tuples in D for which A split point, while D_2holds the rest.

A is discrete-valued and a binary tree must be produced (as dictated by the attribute selection measure or algorithm being used): The test at node N is of the form "$A \in S_A$," where S_A is the splitting subset for A, returned by Attribute selection method as part of the splitting criterion. It is a subset of the known values of A.

Attribute selection measures

Gini Index is used as attribute selection measures in CART. The Gini index measures the impurity of D, a data partition or set of training tuples, as

Gini (D) $= 1 - \sum_{i=1}^{m} Pi^2$

Where p_i is the probability that a tuple in D belongs to class C_i. The Gini index considers a binary split for each attribute. Let's first consider the case where A is a discrete-valued attribute having v distinct values, $a_1, a_2, \ldots, a_v$, occurring in D. To determine the best binary split on A, we examine all the possible subsets that can be formed using known values of A. When considering a binary split, we compute a weighted sum of the impurity of each resulting partition. For example, if a binary split on A partitions D into D_1 and D_2, the Gini index of D given that partitioning is

$$\text{Gini}_A(D) = \frac{D_1}{D} \text{ Gini}(D_1) + \frac{D_2}{D} \text{ Gini } (D_2)$$

The reduction in impurity that would be incurred by a binary split on a discrete- or continuous-valued attribute A is

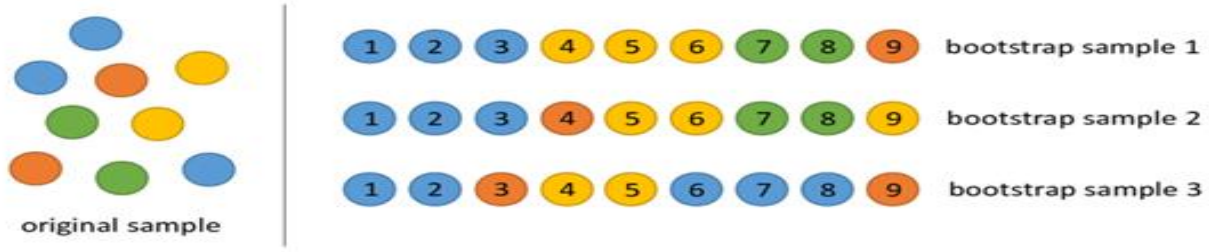
$$\Delta Gini = \text{Gini (D)} - Gini_A(D)$$

The attribute that maximizes the reduction in impurity (or, equivalently, has the minimum Gini index) is selected as the splitting attribute. This attribute and either its splitting subset

(for a discrete-valued splitting attribute) or split-point (for a continuous-valued splitting attribute) together form the splitting criterion.

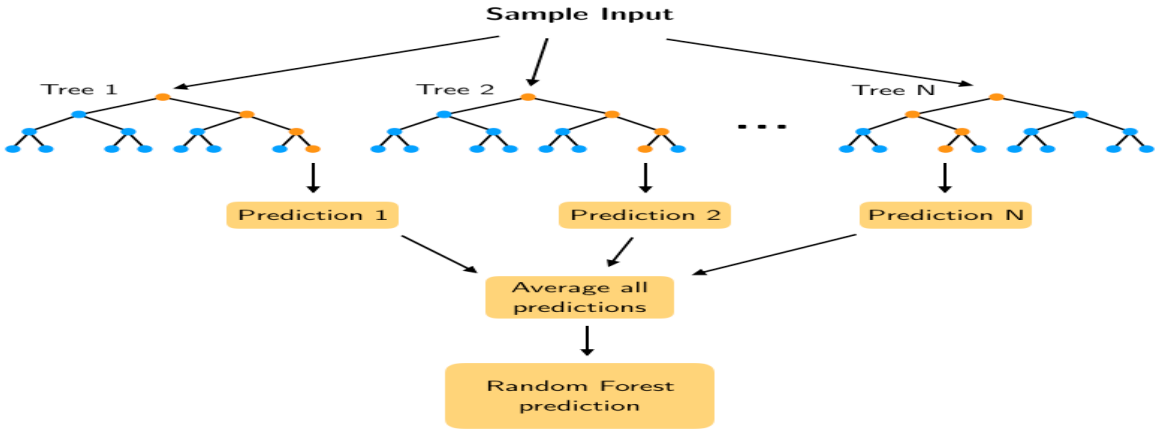
3.6 Random forest

Random forest is a supervised learning algorithm which is used for both classification as well as regression because of its simplicity and high accuracy. Random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. It is an ensemble method which is better than a single decision tree because it reduces the over-fitting by averaging the result. Random Forest algorithm is also very fast and *robust* than other regression models. The success of a random forest highly depends on using uncorrelated decision trees. If we use same or very similar trees, overall result will not be much different than the result of a single decision tree. Random forests achieve to have uncorrelated decision trees by **bootstrapping** and **feature randomness**. Bootstrapping is randomly selecting samples from training data with replacement. They are called bootstrap samples. The following figure clearly explains this process:



Feature randomness is achieved by selecting features randomly for each decision tree in a random forest. The number of features used for each tree in a random forest can be controlled with **max_features** parameter.

For regression, the prediction of a leaf node is the mean value of the target values in that leaf. Random forest regression takes mean value of the results from decision trees.



3.7 K– NN Classification

Nearest-Neighbor classifiers are based on learning by analogy, that is, by comparing a given test tuple with training tuples that are similar to it.When given an unknown tuple, a k-nearest-neighbour classifier searches the pattern space for the k-training tuples that are closest to the unknown tuple.

1. "Closeness" is defined in terms of a distance metric, such as Euclidean distance. The Euclidean distance between two points or tuples, say,

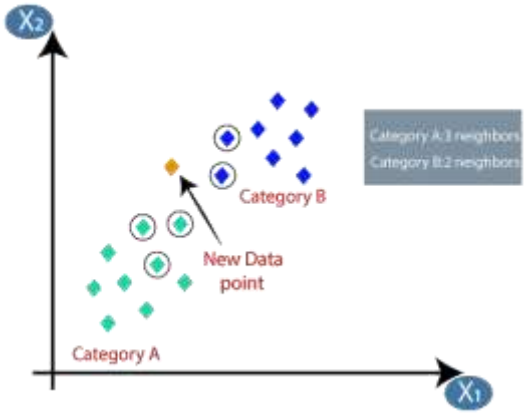
   $X_1=(x_{11},x_{12},\ldots\ldots,x_{1n})$ and $X_2=(x_{21},x_{22},\ldots\ldots,x_{2n})$

   $$dist(X_1,X_2) = \sqrt{(X_{1i} - X_{2i})^2}$$

   We normalize the values of each attribute before using equation. This helps prevent attributes with initially large ranges from outweighing attributes with initially smaller ranges.

2. Min-Max normalization, for example, can be used to transform a value *v* of a numeric attribute A to v^0 in the range [0,1] by computing

$$\mathbf{X_{new}} = \frac{(\mathbf{x-min})}{(\mathbf{max-min})}$$



The 3 nearest neighbours are from category A, hence this new data point must belong to category A.

Determination of good value from k

The good value for k, the no. of nearest neighbour can be determined by the experimental starting with k=1 estimate the error rate of classifier.

The process can be repeated each time by implementing k for one or more valuable and so on.

The value for k for which error rate i.e minimum may be taken as good value for k.

Another method for finding best value of k is given by,

k=$\sqrt{number}\ of\ training\ tuples$

3.8 Support Vector Machine

Support vector machines (SVM) were introduced by Cortes and Vapnik (1995) for classification problems. One year after the introduction of SVM, Smola et al. (1996) advanced the alternative loss function, which also allowed SVM to be applied to regression problems. The objective is to look for the optimal separating hyperplane between classes. The points lying on classes' boundaries are called support vectors, and the in-between space is called the hyperplane; when a linear separator is not able to find a solution, data points are projected into a higher-dimensional space, where the previous nonlinearly separable points become linearly separable, using kernel functions. The whole task can be formulated as a quadratic optimization problem that can be solved with exact techniques. Figure 1 presents an example of a linearly separable classification problem solved using SVM. SVM aims at maximizing the margin between the support vectors and the hyperplane.

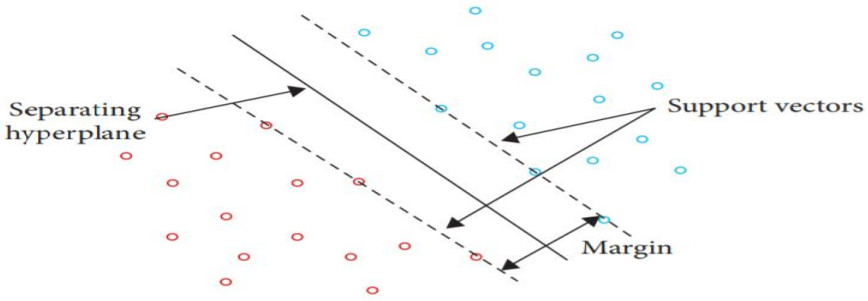


FIGURE 1: Linearly separable problem.

Consider a linear model whose prediction is given by $f(x) = w^T x + b$, where w is the weight vector, b is the bias and x is the input feature vector. Suppose we are given with a training set (x_i, y_i), i = 1, 2, …, n, then the error function is given by

$$J = \frac{1}{2}||w||^2 + c\sum_{i=1}^{n}|y_i - f(x_i)|_\varepsilon$$

The first term in the error function is a term that penalizes model complexity. The second term is the ε-insensitive loss function, can be defined as

$$|y_i - f(x_i)|_\varepsilon = \max\{0, |y_i - f(x_i) - \varepsilon\}$$

It does not penalize errors below ε, allowing the parameters to vary and reduce model complexity. The solution that minimizes the error function can be given by

$$f(x) = \sum_{i=1`}^{n}(\alpha_i^* - \alpha_i)x_i^T x + b$$

where $\alpha_i^*$and α_i are Lagrange multipliers. The training vectors giving non-zero Lagrange multipliers are called support vectors, is a key concept in SVR theory. Non-support vectors do not contribute directly to the solution, and the number of support vectors is a measure of complexity of the model (Cherkassky et al :2004). This model can be extended to the nonlinear case through the concept of kernel K, giving a solution like this:

$$f(x) = \sum_{i=1`}^{n}(\alpha_i^* - \alpha_i)K(x_i^T x) + b$$

A very commonly used kernel function is the Gaussian kernel.

3.9 Naive Bayesian Classifier:

Naive Bayes is a machine learning model that is used for large volumes of data, even if you are working with data that has millions of data records the recommended approach is Naive

Bayes. It gives very good results when it comes to NLP tasks such as sentimental analysis. It is a fast and uncomplicated classification algorithm.

Bayesian classifiers are statistical classifiers. They can predict membership probabilities such as the probability that a given tuple belongs to a particular class.

Bayesian classification is based on Bayes' theorem, described next. Studies comparing classification algorithms have found a simple Bayesian classifier known as the naïve Bayesian classifier to be comparable in performance with decision tree and selected neural network classifiers. Bayesian classifier have also exhibited high accuracy and speed when applied to large database. Naïve Bayesian classifiers assume that the effect of an attribute value on a given class is independent of the values of the other attributes. This assumption is called class-conditional independence. It is made to simplify the computations involved and, in this sense, is considered "naïve". Bayesian belief networks are graphical models, which unlike naïve Bayesian classifiers, allow the representation of dependencies among subsets of attributes. Bayesian belief networks can also be used for classification.

Bayes' theorem:

Let X be a data tuple. Let H be some hypothesis such as that the data tuple X belongs to specified class C. we want to determine P(H|X), ie the probability that tuple X belongs to class C, given that we know that attribute description of X.P(H/X) is the posterior probability, or a posteriori probability, of H conditioned on X. Bayes' theorem is useful in that it provides way of calculating the posterior probability, P(H|X), from P(H), P(X|H) and P(X). Bayes' theorem is

$$P(X|H) = \frac{P(X|H)*P(H)}{P(H)}$$

Let D be training set of tuples and their associated class labels. As usual, each tuple is represented by an n-dimensional attribute vector, X= (x_1, x_2, …, x_n), depicting n measurements made on the tuple from n attributes, respectively A_1, A_2, …A_n.

Suppose that there are m classes, C_1, C_2, …, C_m. Given a tuple, X, the classifier will predict that X belongs to the class having the highest posterior probability, conditioned on X. That is, the native Bayesian classifier predicts that tuple X belongs to the class Ci if and only if $P(C_i|X) > P(C_j|X)$.

Thus, we maximize $P(C_i|X)$. The class Ci for which $P(C_i|X)$ is maximized is called the maximum posteriori hypothesis. By Bayes' theorem

$$p(c_i|x) = \frac{P(X|Ci)*P(Ci)}{P(X)}$$

As P(X) is constant for all classes, only $P(X|C_i) * P(C_i)$ needs to be maximized.

In order to reduce computation in evaluating $P(X|C_i)$, the naïve assumption of class conditional independence is made. This presumes that the attributes' values are conditionally independent of one another, given the class label of the tuple (i.e., there are no dependence relationships among the attributes). Thus,

$$P(X|C_i) = \prod_{k=1}^{n} P(X_k|Ci)$$

$$=(X_1|C_i) * P(X_2|C_i) * \ldots . * P(X_n|C_i)$$

We can easily estimate the probability $P(X_1|C_i)$, $P(X_2|C_i)$, …., $P(X_n|C_i)$ from the training tuples.

To predict the class label of X, $P(X|C_i)$. $P(C_i)$ is evaluated for each class C_i. The classifier predicts that the label of tuple X is the class C_i if and only if

$$(X|C_i) * P(C_i) > P(X|C_j) * P(C_j) \qquad \text{for } 1 \leq j \leq m$$

The predicted class label is the class Ci for which $P(X|C_i) * P(Ci)$ is maximum.

Effectiveness of Bayesian classifier:

Various empirical studies of this classifier in comparison to decision tree and neural network classifiers have found it to be com-parable in some domains. In theory, Bayesian classifier have the minimum error rate in comparison to all other classifiers. However, in practice this is not always the case, owing to inaccuracies in all the assumptions made for its use, then such as class -conditional independence, and the lack of available probability data.

Bayesian classifiers are also useful in that they provide a theoretical justification for other classifiers that do not explicitly use Bayes' theorem. For example, under certain assumptions, it can be shown that many neural network and curve-fitting algorithms output the maximum posteriori hypothesis, as does the naïve Bayesian classifier.

3.10 Linear Discriminant Analysis

Linear discriminant analysis (LDA) and Fisher's linear discriminant are methods used in statistics, pattern recognition and machine learning to find a linear combination of features which characterize or separate two or more classes of objects or events. The resulting combination may be used as a linear classifier, or, more commonly, for dimensionality reduction before later classification. LDA is also closely related to principal component analysis (PCA) and factor analysis in that both look for linear combinations of variables which best explain the data. LDA explicitly attempts to model the difference between the classes of data. PCA on the other hand does not take into account any difference in class, and factor analysis builds the feature combinations based on differences rather than similarities. Discriminant analysis is also different from factor analysis in that it is not an interdependence technique.

The objective of LDA is to perform dimensionality reduction while preserving as much of the class discriminatory information as possible.

Assume we have a set of D-dimensional samples $\{x_1, x_2 \ldots x_N\}$, N_1 of which

belong to class ω_1, and N_2 to class ω_2. We seek to obtain a scalar y by projecting the samples x onto a line

$$y = w^T x$$

Of all the possible lines we would like to select the one that maximizes the separability of the scalars.

The mean vector of each class in x and y feature space is

$$\mu i = \frac{1}{Ni} \sum X \epsilon \omega_i X$$

$$\hat{\mu} = \frac{1}{Ni} \sum Y \varepsilon \omega_i Y = \frac{1}{Ni} \sum X \varepsilon \omega_i W^T X = W^T \mu i$$

We could then choose the distance between the projected means as our objective function.

$$J(w) = |\hat{\mu}_1 - \hat{\mu}_2| = |W^T(\mu_1 - \mu_2)|$$

However, the distance between the projected means is not a very good measure since it does not take into account the standard deviation within the classes. The solution proposed by Fisher is to maximize a function that represents the difference between the means, normalized by a measure of the within - class scatter.

For each class we define the scatter, an equivalent of the variance, as

$$\hat{S}_l^{\,2} = \sum Y \varepsilon \omega_i \, (y - \hat{\mu}_l)^2$$

where the quantity $\left(\hat{S}_1^{\,2} + \hat{S}_2^{\,2}\right)$ is called the within-class scatter of the projected examples The Fisher linear discriminant is defined as the linear function $w^T x$ that maximizes the criterion function.

$$J(w) = \frac{|\hat{\mu}_1 - \hat{\mu}_2|^2}{\left(\hat{s}_1^{\,2} + \hat{s}_2^{\,2}\right)}$$

Therefore, we will be looking for a projection where examples from the same class are projected very close to each other and, at the same time, the projected means are as farther apart as possible.

In order to find the optimum projection w*, we need to express J(w) as an explicit function of w.

We define a measure of the scatter in multivariate feature space x, which are scatter matrices

$$S_i = \sum X \varepsilon \omega_i \, (x - \mu_i)^T$$

$$S_1 + S_2 = S_w$$

Where S_W is called the within-class scatter matrix.

The scatter of the projection y can then be expressed as a function of the scatter matrix in feature space x

$$\hat{S}_l^{\,2} = \sum Y \varepsilon \omega_i \, (y - \hat{\mu}_l)^2 = \sum X \varepsilon \omega_i \, (w^T x - w^T \mu_i)^2$$

$$= \sum X \varepsilon \omega_i \, w^T (x - \mu_i)^T w$$

$$= w^T S_i w$$

$$= \hat{S}_1^{\,2} + \hat{S}_2^{\,2} = w^T S_w w$$

Similarly, the difference between the projected means can be expressed in terms of the means in the original feature space.

$$(\hat{\mu}_1 - \hat{\mu}_2)^2 = (w^T\mu_1 - w^T\mu_2)^2$$

$$w^T(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T w = w^T S_B w$$

The matrix SB is called the between-class scatter. Note that, since SB is the outer product of two vectors, its rank is at most one.

We can finally express the Fisher criterion in terms of S_W and S_B as

$$J(w) = \frac{w^T S_B w}{w^T S_w w}$$

To find the maximum of J (w) we derive and equate to zero.

$$\frac{d}{dw}[J(w)] = \frac{d}{dw}\left[\frac{w^T S_B w}{w^T S_w w}\right] = 0 \Rightarrow$$

Dividing by $w^T S_w w$

$$\left[\frac{w^T S_B w}{w^T S_w w}\right] S_B w - \left[\frac{w^T S_B w}{w^T S_w w}\right] S_w w = 0 \Rightarrow$$

$$\Rightarrow S_B w - J S_w w = 0 \Rightarrow$$

$$\Rightarrow S_w^{-1} S_B w - J(w) = 0$$

Solving the generalized eigenvalue problem $(S_w{}^{-1} S_B w = \mathrm{J}(w))$ yields

$$w^* = \arg\max_w \left\{\frac{w^T S_B w}{w^T S_w w}\right\} = S_w^{-1}(\mu_1 - \mu_{2l})$$

This is known as Fisher's Linear Discriminant, although it is not a discriminant but rather a specific choice of direction for the projection of the data down to one dimension.

3.11 Bagging

Bagging, also known as bootstrap aggregating, is the aggregation of multiple versions of a predicted model. Each model is trained individually, and combined using an averaging process. The primary focus of bagging is to achieve less variance than any model has individually.

It all started in the year 1994, when Leo Breiman proposed this algorithm, then known as "Bagging Predictors". In Bagging, the bootstrapped samples are first created. Then, either a regression or classification algorithm is applied to each sample. Finally, in the case of regression, an average is taken over all the outputs predicted by the individual learners. For classification either the most voted class is accepted (hard-voting), or the highest average of all the class probabilities is taken as the output (soft-voting). This is where aggregation comes into the picture.

Bagging works especially well when the learners are unstable and tend to overfit, i.e. small changes in the training data lead to major changes in the predicted output. It effectively reduces the variance by aggregating the individual learners composed of different statistical properties, such as different standard deviations, means, etc. It works well for high variance models such as Decision Trees. When used with low variance models such as linear regression, it doesn't really affect the learning process. The number of base learners (trees) to be chosen depends on the characteristics of the dataset. Using too many trees doesn't lead to overfitting, but can consume a lot of computational power.

Bagging can be done in parallel to keep a check on excessive computational resources. This is a one good advantages that comes with it, and often is a booster to increase the usage of the algorithm in a variety of areas.

The step-by-step procedure that goes into implementing the Bagging algorithm.

- Bootstrapping comprises the first step in bagging process flow wherein the data is divided into randomized samples.
- Then fit another algorithm (e.g. Decision Trees) to each of these samples.

  Training happens in parallel.

- Take an average of all the outputs, or in general, compute the aggregated output.

Confusion matrix for a binary classifier:

	Predicted Values	
Actual Values	***NO (0)***	***YES (1)***
NO (0)	**True Negative**	**False Positive**
YES (1)	**False Negative**	**True Positive**

There are two possible predicted classes: "yes" and "no". If we were predicting the presence of a disease, for example, "yes" would mean they have the disease, and "no" would mean they don't have the disease.

Let's now define the most basic terms, which are whole numbers (not rates):

- **True positives (TP):** These are cases in which we predicted yes (they have the disease), and they do have the disease.
- **True negatives (TN):** We predicted no, and they don't have the disease.
- **False positives (FP):** We predicted yes, but they don't actually have the disease. (Also known as a "Type I error.")
- **False negatives (FN):** We predicted no, but they actually do have the disease. (Also known as a "Type II error.")

This is a list of rates that are often computed from a confusion matrix for a binary classifier:

1. **Accuracy:** Overall, how often is the classifier correct?

   Accuracy = (TP+TN)/total

2. **Misclassification Rate:** Overall, how often is it wrong?

   Misclassification Rate = (FP+FN)/total

   It is equivalent to 1 minus Accuracy and also known as "Error Rate".

3. **True Positive Rate:** When it's actually yes, how often does it predict yes? Also known as "Sensitivity" or "Recall"

   Sensitivity = TP/total positive

4. **True Negative Rate:** When it's actually no, how often does it predict no? Also known as "Specificity"

   Specificity = TN/total negative

Evaluation Measures:

Sensitivity = TP / (TP + FN)

Specificity = TN / (TN + FP)

Accuracy = (TP + TN) / Total

CHAPTER-4

ANALYSIS AND DISCUSSION

Descriptive statistics

Descriptive statistics for non-categorical variables included in the study

Variables	Minimum	Mean	Median	Skewness	Kurtosis	Standard Deviation	Maximum
birth	81.684	84.708743	84.8855	-0.166099	-1.05085	1.589532	87.483
exit	81.046	84.855872	85.0115	-0.188898	-0.9805	1.596442	87.718
hospitalstay	2	46.748691	45	0.533743	-0.23253	26.92957	123
lowph	6.889999	7.212799	7.209999	-0.273406	-0.03345	0.119811	7.549999
pltct	16	199.468586	203.25	0.059825	-0.25164	77.15775	399
bwt	400	1108.62827	1120	-0.288092	-0.7245	237.35369	1500
gest	23	29.081152	29	0.355329	-0.07062	2.1571977	36
apg1	0	4.955497	5.5	-0.183144	-1.36447	2.6808563	9

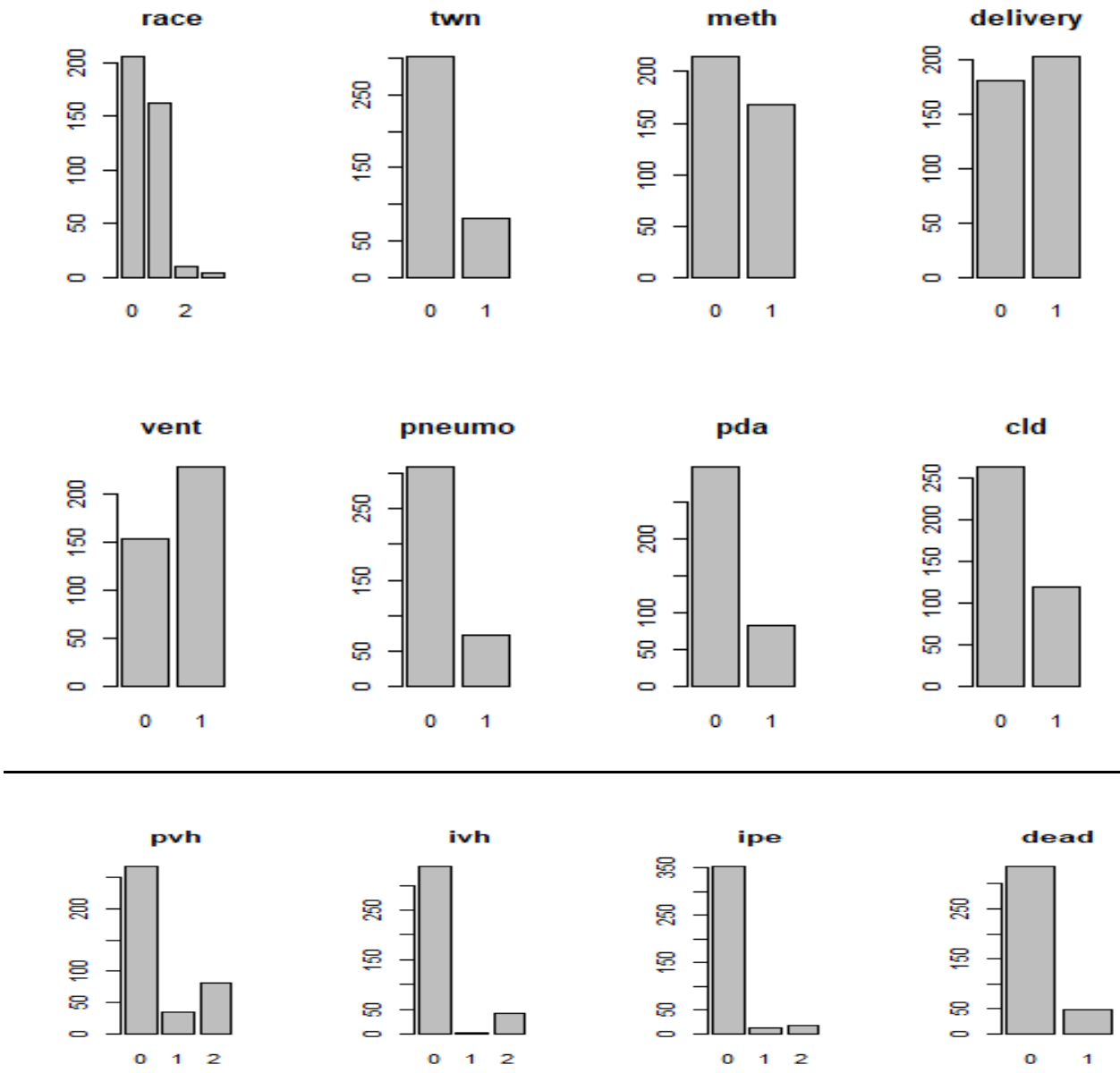
Table 1

From table 1, we can observe that birth, lowph, Birth weight in gram, Apgar and exit variables are negatively skewed and variables such as hospital stay, plate late count and Gestational age in week are positively skewed. Also observed that kurtosis value of all variables are less than 3, we say that is plattykurtic.

Mode of categorical variables included in the study:

Variables	Race	twn	meth	delivery	vent	pneumo	Pda	cld	pvh	ivh	ipe	dead
Mode	0	0	0	1	1	0	0	0	0	0	0	0

BAR PLOT: Bar plot for categorical data is given below.



From the bar plot, we observe that most of the neonates are surviving, we can observe most of the number black infant in the data set, number of multiple gestation very low, Mother treated with beta-methasone is low ,we can observe that there are more number of normal delivery than abdominal, we can see more number of Assisted ventilation used, Pneumothorax occurred is less in the data set, in the data Patent ductus arteriosus detected is low, we can observe

that more number of infants without Periventricular haemorrhage, Intraventricular haemorrhage, Periventricular intraparenchymal echodense lesion.

Imbalanced data:

Logistic Regression

We use multivariate logistic regression model to find the risk factors associate with Low Birth Weight.

Table 1: Fitted logistic regression model for original data

AIC: 85.53

Coefficients:

	Estimate	Std. Error t	value	(Pr >|t|)
(Intercept)	3.85E+00	1.27E+00	3.031	0.002613
birth	-2.85E-02	8.17E-02	-0.349	0.727156
exit	-4.87E-03	8.14E-02	-0.06	0.952375
hospstay	-6.17E-03	6.57E-04	-9.395	< 2e-16
lowph	2.62E-02	1.52E-01	0.172	0.86319
pltct	-1.44E-04	1.97E-04	-0.732	0.464665
r2	7.33E-02	2.82E-02	2.599	0.009748
r3	1.59E-01	8.73E-02	1.82	0.06961
r4	-2.11E-01	1.54E-01	-1.37	0.171701
bwt	-3.68E-04	8.52E-05	-4.322	2.01E-05
gest	-1.78E-02	9.31E-03	-1.913	0.056517
twn	5.09E-03	3.68E-02	0.138	0.890115
meth	1.27E-02	3.17E-02	0.399	0.690338
toc	3.31E-02	3.41E-02	0.972	0.331592
delivery	-3.54E-02	2.88E-02	-1.23	0.219687
apg1	-3.38E-03	5.80E-03	-0.583	0.560053
vent	9.80E-02	3.87E-02	2.532	0.011757
pneumo	1.89E-01	3.95E-02	4.783	2.53E-06
pda	1.52E-01	3.94E-02	3.856	0.000137
cld	3.47E-02	4.17E-02	0.833	0.405263
pvh2	6.70E-02	4.82E-02	1.39	0.165535
pvh3	-5.32E-02	3.43E-02	-1.551	0.121738
ivh2	-1.14E-01	1.56E-01	-0.729	0.466752
ivh3	-6.17E-03	4.45E-02	-0.139	0.889862
ipe2	-3.35E-03	7.64E-02	-0.044	0.96506
ipe3	-3.70E-03	6.62E-02	-0.056	0.955469
sex	2.04E-02	2.83E-02	0.721	0.47161

From the above table, we can observe that the ‘pneumo’, ‘pda’, ‘vent, ‘gest’, ‘bwt’,’ r3’,‘pltct’ and ‘hospstay’ are the significant variables for the study.

Table 2: Stepwise Logistic Regression

We have applied backward stepwise logistic regression to select the best subsets. Based on the minimum AIC value we choose the best combination of variables.

Start: AIC = 85.53

dead ~ birth + exit + hospstay + lowph + pltct + r2 + r3 + r4 +bwt + gest + twn + meth + toc + delivery + apg1 + vent + pneumo + pda + cld + pvh2 + pvh3 + ivh2 + ivh3 + ipe2 + ipe3

Coefficients

	Df	Deviance	AIC
ipe2	1	24.163	83.528
ipe3	1	24.163	83.529
exit	1	24.164	83.53
twn	1	24.165	83.546
ivh3	1	24.165	83.546
lowph	1	24.165	83.558
birth	1	24.172	83.657
meth	1	24.174	83.697
apg1	1	24.186	83.892
sex	1	24.199	84.084
ivh2	1	24.2	84.096
pltct	1	24.2	84.102
cld	1	24.211	84.272
toc	1	24.228	84.541
delivery	1	24.266	85.149
none>		24.163	85.526
r4	1	24.291	85.539
pvh2	1	24.295	85.598
pvh3	1	24.327	86.106
r3	1	24.389	87.073
gest	1	24.413	87.445
vent	1	24.6	90.365
r2	1	24.623	90.724
pda	1	25.175	99.198
bwt	1	25.435	103.119

pneumo	1	25.721	107.384
hospstay	1	30.172	168.357

Step 1: AIC = 83.53

dead ~ birth + exit + hospstay + lowph + pltct + r2 + r3 + r4 +bwt + gest + twn + meth + toc + delivery + apg1 + vent + pneumo + pda + cld + pvh2 + pvh3 + ivh2 + ivh3 + sex

Coefficients:

	Df	Deviance	AIC
exit	1	24.164	79.535
twn	1	24.165	79.553
ivh3	1	24.165	79.554
lowph	1	24.166	79.563
birth	1	24.172	79.664
meth	1	24.174	79.703
apg1	1	24.187	79.896
sex	1	24.199	80.085
ivh2	1	24.2	80.098
pltct	1	24.2	80.107
cld	1	24.211	80.276
toc	1	24.228	80.545
delivery	1	24.267	81.156
none>		24.164	81.531
r4	1	24.291	81.542
pvh2	1	24.295	81.599
pvh3	1	24.328	82.12
r3	1	24.39	83.084
gest	1	24.417	83.519
vent	1	24.604	86.428
r2	1	24.627	86.779
pda	1	25.182	95.303
bwt	1	25.437	99.152
pneumo	1	25.744	103.738
hospstay	1	30.194	164.636

Step 2: AIC = 81.53

dead ~ birth + exit + hospstay + lowph + pltct + r2 + r3 + r4 + bwt + gest + twn + meth + toc + delivery + apg1 + vent + pneumo + pda + cld + pvh2 + pvh3 + ivh2 + ivh3 + sex

Similarly repeating same procedure and by removing the variables which has less information, finally we got following result:

Step14: AIC=62.65

dead ~ birth + hospstay + r2 + r3 + r4 + bwt + gest + vent +pneumo + pda + pvh2 + pvh3

coefficients:

	Df	Deviance	AIC
none>		24.49	62.654
pvh3	1	24.619	62.661
r4	1	24.631	62.852
pvh2	1	24.643	63.031
r3	1	24.718	64.188
gest	1	24.851	66.251
r2	1	24.913	67.191
vent	1	25.076	69.682
birth	1	25.529	76.533
pda	1	25.841	81.165
bwt	1	26.005	83.582
pneumo	1	26.223	86.769
hospstay	1	31.809	160.545

From stepwise regression we observe birth , hospstay, race2, race4, race3, bwt, gest, vent, pneumo, pda, pvh2, pvh3 are the significant variables.

By removing the insignificant variables from the model we have fitted a new model for the significant variables.

Table 3: Fit the model after getting significant variables

Coefficients:

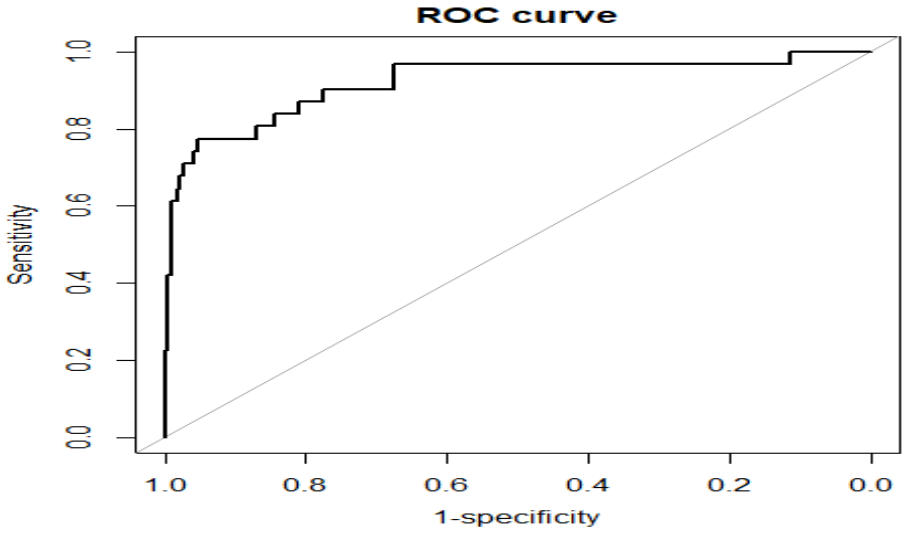
	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	4.25E+00	7.74E-01	5.492	7.41E-08
Birth	-3.54E-02	8.94E-03	-3.958	9.08E-05
Hospstay	-5.99E-03	5.70E-04	-10.502	< 2e-16
r2	6.89E-02	2.73E-02	2.524	0.01203
r3	1.58E-01	8.53E-02	1.852	0.06481
Bwt	-3.75E-04	7.86E-05	-4.778	2.56E-06
Gest	-2.02E-02	8.67E-03	-2.334	0.02014
Vent	9.68E-02	3.26E-02	2.971	0.00317
Pneumo	1.89E-01	3.71E-02	5.11	5.19E-07
Pda	1.63E-01	3.60E-02	4.512	8.66E-06

These are all the significant variables for our study and henceforth we'll be using only these variables for the analysis

The fitted logistic model is given by

$\log\left(\frac{\pi(x)}{1-\pi(x)}\right)$ =4.2489258+ -0.0353795 birth -0.0059848 hospstay + 0.0688703r2+ 0.1579239 r3 - 0.0003753 bwt -0.0202381 gest +0.0967708 vent +0.1894236 pneumo + 0.1625272 pda

ROC curve



Area under the curve: 0.9205

Here ROC curve which goes close to the top left corner of the plot, indicates the high discrimination ability of the fitted model. Area under the ROC curve is 0.9205, which strongly suggests that the model has a satisfactory discriminating power.

This means that the test is 92% good in a given clinical situation. The test separates the data into two groups (survive or not) with 92% accuracy

❖ Classification using Logistic Regression

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	94	3
Class 1	4	13

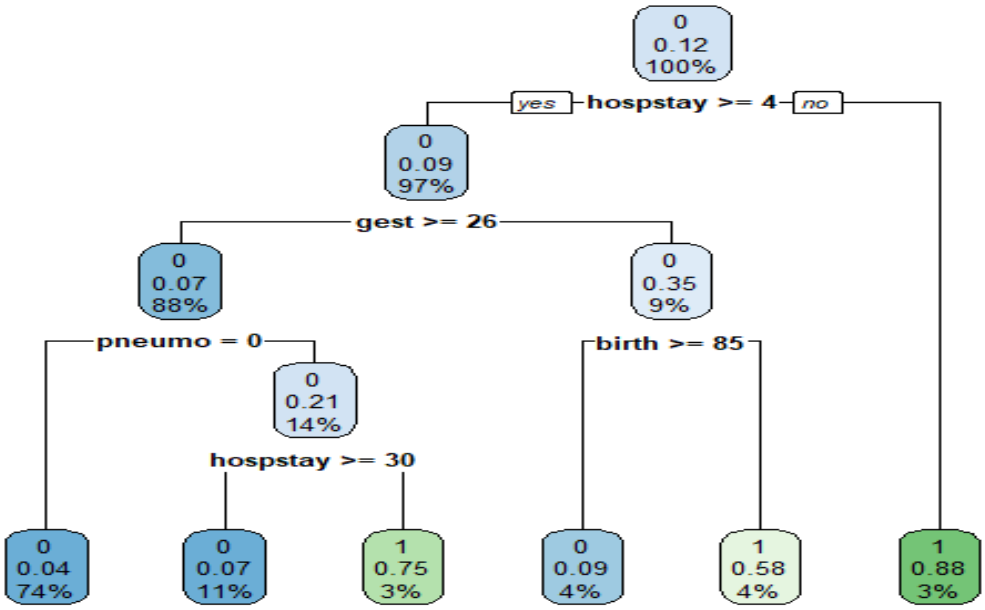
From the above table, we observe that

1. Out of 97 normal Babies, 94 are classified as they are alive, but 3 Babies are misclassified as they are dead.
2. Out of 17 Babies, 13 are correctly classified as they are alive and 4 are misclassified.

Accuracy	Sensitivity	Specificity
93.85965%	96.90722%	76.47059%

- From 114 observations, 94.85% are correctly classified under the respective classes. 6.15 % of observations are misclassified.
- The Sensitivity of the Logistic Regression is 96.90722% that means the test is correctly classified as those who are Alive.
- The Specificity of the Logistic Regression is 76.47059% , that means the test is correctly classified as those who are Dead.

❖ Classification using Decision Tree:



According to the above decision tree hospital stay, Gestational age in weeks, pneumonia, birth, are important variables which are the reasons for very low birth weight infant.

Table 1: Confusion Matrix and Accuracy

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	91	6
Class 1	14	3

From the above table, we observe that

1. Out of 97 normal Babies 91 are classified as they are alive, but 6 Babies are misclassified as they are dead.
2. Out of 17 Babies 3 are correctly classified as they are alive and 14 are misclassified.

Accuracy	Sensitivity	Specificity
82.45614%	93.81443%	17.64706%

- From 114 observations, 82.45% are correctly classified under the respective classes. 17.55 % of observations are misclassified
- The Sensitivity of the Decision Tree is 96.90722% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Decision Tree is 17.64706%, that means the test is correctly classified as those who are Dead

❖ Classification using Random Forest:

Table 1: Confusion Matrix and Accuracy

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	97	0
Class 1	12	5

From the above table, we observe that

1. Out of 97 normal Babies 97 are classified as they are alive
2. Out of 17 Babies 5 are correctly classified as they are alive and 12 are misclassified.

Accuracy	Sensitivity	Specificity
89.47368 %	100 %	17.64706%

- From 114 observations, 89.47% are correctly classified under the respective classes. 10.53 % of observations are misclassified
- The Sensitivity of the Random Forest is 100 % , that means the test is correctly classified as those who are Alive.
- The Specificity of the Random Forest is 29.41176 %, that means the test is correctly classified as those who are Dead

❖ Classification using K-NN classification

Table 1: Confusion Matrix and Accuracy

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	91	6
Class 1	13	4

From the above table, we observe that

1. Out of 97 normal Babies 91 are classified as they are alive, but 6 Babies are misclassified as they are dead.
2. Out of 17 Babies 4 are correctly classified as they are alive and 13 are misclassified.

Accuracy	Sensitivity	Specificity
83.33333%	93.81443%	23.52941%

- From 114 observations, 83.33% are correctly classified under the respective classes. 16.67 % of observations are misclassified.
- The Sensitivity of the K-NN Classification is 93.81443% , that means the test is correctly classified as those who are Alive.
- The Specificity of the K-NN Classification 23.52941% , that means the test is correctly classified as those who are Dead.

❖ Classification using Support Vector Machine

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	96	1
Class 1	10	7

From the above table, we observe that

1. Out of 97 normal Babies 96 are classified as they are alive, but 1 Babies are misclassified as they are dead.
2. Out of 17 Babies 7 are correctly classified as they are alive and 10 are misclassified.

Accuracy	Sensitivity	Specificity
90.35088%	98.96907%	41.17647%

- From 114 observations, 90.35% are correctly classified under the respective classes. 9.65 % of observations are misclassified
- The Sensitivity of the Support Vector machine is 98.96907% **,** that means the test is correctly classified as those who are Alive.
- The Specificity of the Support Vector machine is 41.17647% **,** that means the test is correctly classified as those who are Dead.

❖ Naive Baye's Classification

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	91	6
Class 1	2	15

From the above table, we observe that

1. Out of 97 normal Babies 91 are classified as they are alive, but 6 Babies are misclassified as they are dead.
2. Out of 17 Babies 15 are correctly classified as they are alive and 2 are misclassified.

Accuracy	Sensitivity	Specificity
92.98246%	93.81443%	88.23529%

- From 114 observations, 92.98% are correctly classified under the respective classes. 7.02 % of observations are misclassified.
- The Sensitivity of the Naïve Bayes Classifier is 93.81443% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Naïve Bayes Classifier 88.23529% , that means the test is correctly classified as those who are Dead

❖ Classification using Linear discriminant analysis

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	94	3
Class 1	4	13

From the above table, we observe that

1. Out of 97 normal Babies 94 are classified as they are alive, but 3 Babies are misclassified as they are dead.
2. Out of 17 Babies 13 are correctly classified as they are alive and 4 are misclassified.

Accuracy	Sensitivity	Specificity
93.85965%	96.9072%	76.47059%

- From 114 observations, 93.85% are correctly classified under the respective classes. 6.15 % of observations are misclassified.
- The Sensitivity of the Linear Discriminant Analysis is 96.9072% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Linear Discriminant Analysis 76.47059% , that means the test is correctly classified as those who are Dead

❖ Classification using Bagging:

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	92	5
Class 1	8	9

From the above table, we observe that

1. Out of 97 normal Babies 92 are classified as they are alive, but 5 Babies are misclassified as they are dead.
2. Out of 17 Babies 9 are correctly classified as they are alive and 8 are misclassified.

Accuracy	Sensitivity	Specificity
88.59649%	94.84536%	52.94118%

- From 114 observations, 88.59% are correctly classified under the respective classes. 11.41% of observations are misclassified.
- The Sensitivity of the Bagging is 94.84536% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Bagging 52.94118% , that means the test is correctly classified as those who are Dead.

Synthetic Minority Oversampling Technique:

It aims to balance class distribution by randomly increasing minority class using synthetic training records are generated by randomly selecting one or more of the k-nearest neighbors for each example in the minority class. Under this sampling technique, we use the same classification models as we considered for imbalanced data.

❖ Classification using Logistic Regression

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	86	13
Class 1	15	87

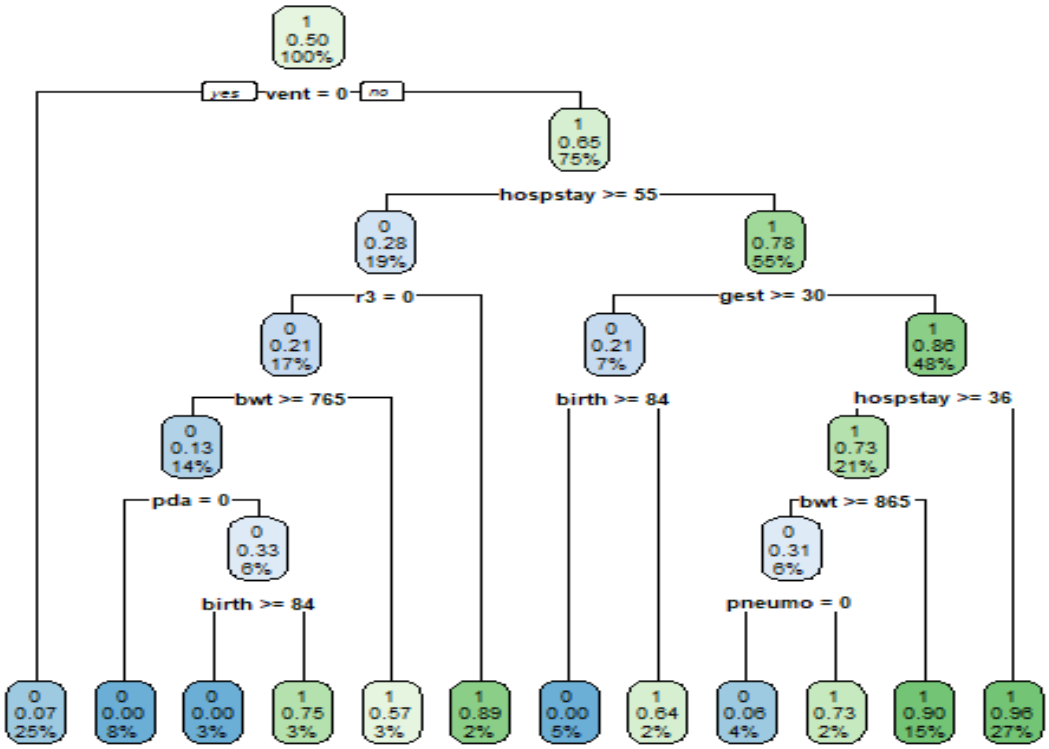
From the above table, we observe that

1. Out of 99 normal Babies 86 are classified as they are alive, but 13 Babies are misclassified as they are dead.
2. Out of 102 Babies 87 are correctly classified as they are alive and 15 are misclassified.

Accuracy	Sensitivity	Specificity
86.06965%	86.86869%	85.29412%

- From 201 observations, 86.06% are correctly classified under the respective classes. 13.94 % of observations are misclassified
- The Sensitivity of the Logistic regression is 86.86869% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Logistic regression 85.29412% , that means the test is correctly classified as those who are Dead

❖ Classification using Decision Tree



According to the above decision tree hospital stay, Patent ductus arteriosus , Gestational age in weeks, pneumonia , Assisted ventilation , birth, Birth weight in gram, race are important variables which are the reasons for very low birth weight infant.

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	79	20
Class 1	3	99

From the above table, we observe that

1. Out of 99 normal Babies 79 are classified as they are alive, but 20 Babies are misclassified as they are dead.

2. Out of 102 Babies 99 are correctly classified as they are alive and 3 are misclassified.

Accuracy	Sensitivity	Specificity
88.55721%	79.79798%	97.05882%

- From 201 observations, 88.55% are correctly classified under the respective classes. 11.45 % of observations are misclassified
- The Sensitivity of the Decision Tree is 79.79798% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Decision Tree 97.05882% , that means the test is correctly classified as those who are Dead.

❖ Classification using Random Forest:

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	96	3
Class 1	3	99

From the above table, we observe that

1. Out of 99 normal Babies 96 are classified as they are alive, but 3 Babies are misclassified as they are dead.
2. Out of 102 Babies 99 are correctly classified as they are alive and 3 are misclassified.

Accuracy	Sensitivity	Specificity
97.01493%	96.9697%	97.05882%

- From 201 observations, 97.014% are correctly classified under the respective classes. 2.99 % of observations are misclassified

- The Sensitivity of the Random Forest is 96.9697% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Random Forest is 97.05882% , that means the test is correctly classified as those who are dead.

❖ Classification using K-NN classification

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	68	31
Class 1	0	102

From the above table, we observe that

1. Out of 99 normal Babies 68 are classified as they are alive, but 31 Babies are misclassified as they are dead.
2. Out of 102 Babies 102 are correctly classified as they are alive.

Accuracy	Sensitivity	Specificity
84.57711%	68.68687%	100%

- From 201 observations, 84.57% are correctly classified under the respective classes. 15.43% of observations are misclassified.
- The Sensitivity of the K-NN Classification is 68.68687% , that means the test is correctly classified as those who are Alive.
- The Specificity of the K-NN Classification is 100% , that means the test is correctly classified as those who are Dead

❖ Classification using Support Vector Machine

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	91	8
Class 1	8	94

From the above table, we observe that

1. Out of 99 normal Babies 91 are classified as they are alive, but 8 Babies are misclassified as they are dead.
2. Out of 102 Babies 94 are correctly classified as they are alive and 8 are misclassified.

Accuracy	Sensitivity	Specificity
92.0398%	91.91919%	92.15686%

- From 201 observations, 92.03% are correctly classified under the respective classes. 7.97 % of observations are misclassified.
- The Sensitivity of the Support Vector Machine is 91.91919% **,** that means the test is correctly classified as those who are Alive.
- The Specificity of the Support Vector Machine is 92.15686% **,** that means the test is correctly classified as those who are Dead

❖ Naive Bayes Classification

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	72	27
Class 1	12	90

From the above table, we observe that

1. Out of 99 normal Babies 72 are classified as they are alive, but 27 Babies are misclassified as they are dead.
2. Out of 102 Babies 90 are correctly classified as they are alive and 13 are misclassified.

Accuracy	Sensitivity	Specificity
80.59701%	80.59701%	88.23529%

- From 201 observations, 80.59% are correctly classified under the respective classes. 19.41 % of observations are misclassified.
- The Sensitivity of the Naïve Bayes Classifier is 80.59701% **,** that means the test is correctly classified as those who are Alive.
- The Specificity of the Naïve Bayes Classifier is 88.23529% **,** that means the test is correctly classified as those who are Dead.

❖ Classification using Linear discriminant analysis

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	87	12
Class 1	15	87

From the above table, we observe that

1. Out of 99 normal Babies 87 are classified as they are alive, but 12 Babies are misclassified as they are dead.
2. Out of 102 Babies 87 are correctly classified as they are alive and 15 are misclassified.

Accuracy	Sensitivity	Specificity
86.56716%	87.87879%	85.29412%

- From 201 observations, 86.56% are correctly classified under the respective classes. 13.44 % of observations are misclassified.
- The Sensitivity of the Linear Discriminant Analysis is 87.87879% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Linear Discriminant Analysis 85.29412% , that means the test is correctly classified as those who are Dead

❖ Classification using Bagging

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	90	9
Class 1	3	99

From the above table, we observe that

1. Out of 99 normal Babies 90 are classified as they are alive, but 9 Babies are misclassified as they are dead.
2. Out of 102 Babies 99 are correctly classified as they are alive and 3 are misclassified.

Accuracy	Sensitivity	Specificity
94.02985%	90.90909%	97.05882%

- From 201 observations, 94.02% are correctly classified under the respective classes. 5.98% of observations are misclassified.
- The Sensitivity of the Bagging is 90.90909% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Bagging is 97.05882% , that means the test is correctly classified as those who are Dead.

- ## Random Over-Sampling technique:

It handles both continuous and categorical data by generating synthetic examples from a conditional density estimate of the two classes. It uses smoothed bootstrapping to draw artificial samples from the feature space neighbourhood around the minority class. Under this sampling technique, we use the same classification models as we considered for imbalanced data.

❖ Classification using Logistic Regression

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	47	13
Class 1	12	42

From the above table, we observe that

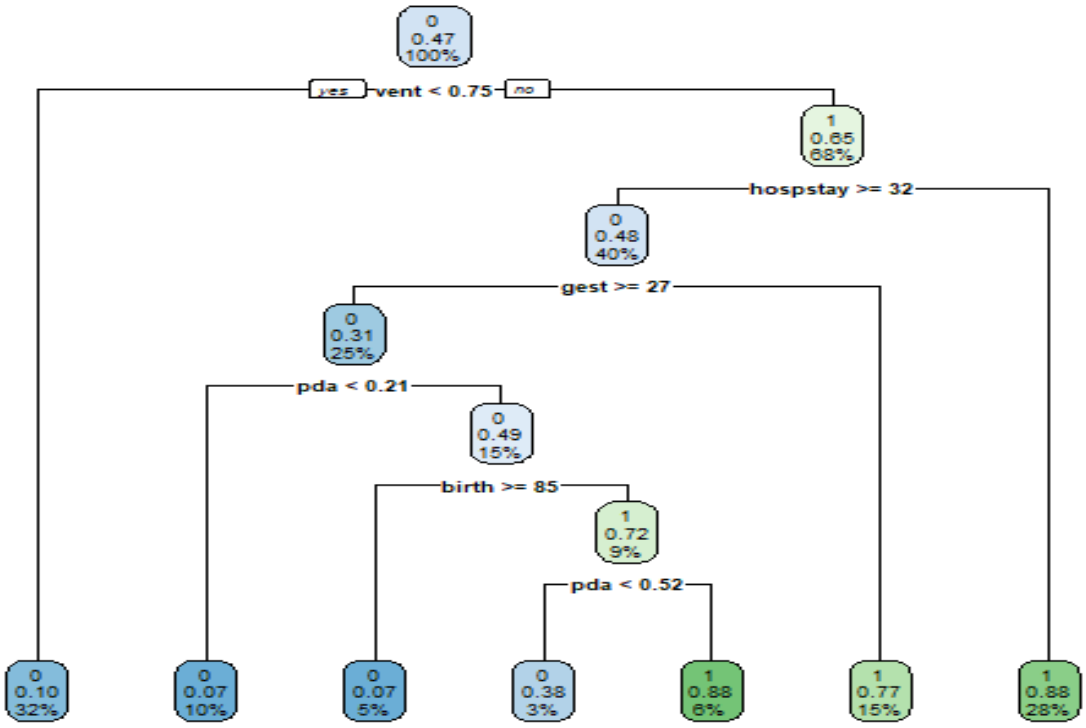
1. Out of 60 normal Babies 47 are classified as they are alive, but 13 Babies are misclassified as they are dead.

2. Out of 54 Babies 42 are correctly classified as they are alive and 12 are misclassified.

Accuracy	Sensitivity	Specificity
78.07018%	78.33333%	77.77778%

- From 114 observations, 78.07.85% are correctly classified under the respective classes. 21.93 % of observations are misclassified.
- The Sensitivity of the Logistic Regression is 78.33333% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Logistic Regression is 77.77778% , that means the test is correctly classified as those who are Dead**.**

❖ Classification using Decision Tree



According to the above decision tree hospital stay, Gestational age in weeks, Assisted ventilation used , birth, Patent ductus arteriosus are important variables which are the reasons for very low birth weight infant.

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	45	15
Class 1	11	43

From the above table, we observe that

1. Out of 60 normal Babies 45 are classified as they are alive, but 15 Babies are misclassified as they are dead.

2. Out of 54 Babies 43 are correctly classified as they are alive and 11 are misclassified.

Accuracy	Sensitivity	Specificity
77.19298%	75%	79.62963%

- From 114 observations, 77.19% are correctly classified under the respective classes. 22.81 % of observations are misclassified
- The Sensitivity of the Decision Tree is 75% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Decision Tree is 79.62963% , that means the test is correctly classified as those who are Dead

❖ Classification using Random Forest:

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	51	9
Class 1	11	43

From the above table, we observe that

1. Out of 60 normal Babies 51 are classified as they are alive, but 9 Babies are misclassified as they are dead.
2. Out of 54 Babies 43 are correctly classified as they are alive and 11 are misclassified.

Accuracy	Sensitivity	Specificity
82.45614%	85%	79.62963%

- From 114 observations, 82.45% are correctly classified under the respective classes. 17.55 % of observations are misclassified

- The Sensitivity of the Random Forest is 85% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Random Forest is 79.62963% , that means the test is correctly classified as those who are Dead

❖ Classification using K-NN classification

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	47	13
Class 1	22	32

From the above table, we observe that

1. Out of 60 normal Babies 47 are classified as they are alive, but 13 Babies are misclassified as they are dead.
2. Out of 54 Babies 32 are correctly classified as they are alive and 22 are misclassified.

Accuracy	Sensitivity	Specificity
69.29825%	78.33333%	59.25926%

- From 114 observations, 69.29% are correctly classified under the respective classes. 30.71 % of observations are misclassified.
- The Sensitivity of the K-NN Classification is 78.33333% , that means the test is correctly classified as those who are Alive.
- The Specificity of the K-NN Classification is 59.25926% , that means the test is correctly classified as those who are Dead.

❖ Classification using Support Vector Machine

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	55	5
Class 1	11	43

From the above table, we observe that

1. Out of 60 normal Babies 55 are classified as they are alive, but 5 Babies are misclassified as they are dead.
2. Out of 54 Babies 43 are correctly classified as they are alive and 11 are misclassified.

Accuracy	Sensitivity	Specificity
85.96491%	91.66667%	79.62963%

- From 114 observations, 85.96% are correctly classified under the respective classes. 14.04 % of observations are misclassified.
- The Sensitivity of the Support Vector Machine is 91.66667% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Support Vector Machine is 79.62963% , that means the test is correctly classified as those who are Dead.

❖ Naive Baye's Classification

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	47	13
Class 1	9	45

From the above table, we observe that

1. Out of 60 normal Babies 47 are classified as they are alive, but 13 Babies are misclassified as they are dead.
2. Out of 54 Babies 45 are correctly classified as they are alive and 9 are misclassified.

Accuracy	Sensitivity	Specificity
80.70175%	78.33333%	83.33333%

- From 114 observations, 80.70% are correctly classified under the respective classes. 19.3 % of observations are misclassified.
- The Sensitivity of the Naïve Bayes Classifier is 96.9072% **,** that means the test is correctly classified as those who are Alive.
- The Specificity of the Naïve Bayes Classifier is 83.33333% **,** that means the test is correctly classified as those who are Dead.

❖ Classification using Linear discriminant analysis

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	46	14
Class 1	11	43

From the above table, we observe that

1. Out of 60 normal Babies 46 are classified as they are alive, but 14 Babies are misclassified as they are dead.
2. Out of 54 Babies 43 are correctly classified as they are alive and 11 are misclassified.

Accuracy	Sensitivity	Specificity
78.07018%	76.66667%	79.62963%

- From 114 observations, 78.07% are correctly classified under the respective classes. 21.93 % of observations are misclassified.
- The Sensitivity of the Linear Discriminant Analysis is 76.66667% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Linear Discriminant Analysis is 79.62963% , that means the test is correctly classified as those who are Dead

❖ Classification using Bagging :

Table 1: Confusion Matrix and Accuracy:

Actual Values	Predicted Values	
	Class 0	Class 1
Class 0	47	13
Class 1	8	46

From the above table, we observe that

1. Out of 60 normal Babies 47 are classified as they are alive, but 13 Babies are misclassified as they are dead.
2. Out of 54 Babies 46 are correctly classified as they are alive and 8 are misclassified.

Accuracy	Sensitivity	Specificity
81.57895%	78.33333%	85.18519%

- From 114 observations, 81.57% are correctly classified under the respective classes. 18.43% of observations are misclassified.
- The Sensitivity of the Bagging is 78.33333% , that means the test is correctly classified as those who are Alive.
- The Specificity of the Bagging is 85.18519% , that means the test is correctly classified as those who are Dead.

After fitting various models to the data, the accuracy, sensitivity and specificity are calculated and listed below

For imbalanced data:

Models	Accuracy	Sensitivity	Specificity
logistic regression	93.85965	96.90722	76.47059
decision tree	82.45614	93.81443	17.64706
Random Forest	89.47368	100	29.41176
KNN	83.33333	93.81443	23.52941
SVM	90.35088	98.96907	41.17647
Naive Bayes	92.98246	93.81443	88.23529
L D A	93.85965	96.90722	76.47059
Bagging	88.59649	94.84536	52.94118

For SMOTE data:

Models	Accuracy	Sensitivity	Specificity
logistic regression	86.06965	86.86869	85.29412
decision tree	88.55721	79.79798	97.05882
Random Forest	**97.01493**	**96.9697**	**97.05882**
KNN	84.57711	68.68687	100
SVM	92.0398	91.91919	92.15686
Naive Bayes	80.59701	80.59701	88.23529
L D A	86.56716	87.87879	85.29412
Bagging	94.02985	90.90909	97.05882

For ROSE data:

Models	Accuracy	Sensitivity	Specificity
logistic regression	78.07018	78.33333	77.77778
decision tree	77.19298	75	79.62963
Random Forest	82.45614	85	79.62963
KNN	69.29825	78.33333	59.25926
SVM	85.96491	91.66667	79.62963
Naive Bayes	80.70175	78.33333	83.33333
L D A	78.07018	76.66667	79.62963
Bagging	81.57895	78.33333	85.18519

CHAPTER 5

FINDINGS AND CONCLUSION

In this Project, several applications of statistical procedure are used for the analysis and prediction.

1 We observe that there are 382 babies who had very low birth weight infant, among them 334 babies are survived and 48 are dead. For the categorical variables, we have computed mode and represented them in a bar plot. For the continuous variables, we computed the mean, median standard deviation, skewness and kurtosis. From skewness, we observed that hospital stay, platelet count and gestational age in week are positively skewed and all other variables are negatively skewed. Also observed that all variables under study are platykurtic.

2 Since the dependent variable in the dataset is dichotomous we have applied logistic regression model for the analysis. From logistic regression model, observed that birth, hospital stay, Birth weight in gram, Gestational age in weeks, Assisted ventilation used, Pneumothorax occurred, Patent ductus arteriosus, Race are important variables which are reason for very low birth weight infant. One of the common way to avoid very low birth weight infant is controlling the risk factors, Follow a healthy diet during pregnancy, exercise and maintain a healthy condition**.** Not use alcohol, cigarettes, or illegal drugs**.** From ROC, we observe that fitted model has excellent discrimination power of 92.05%.

3 We have done the classification for imbalanced data, balanced data (From SMOTE and ROSE) and compare them based on evaluation measures.

4 Among them, classifying in very low birth weight infant, considering the accuracy of models, **Random forest** under SOMTE has a high accuracy of **97.01%** with sensitivity **96.96 %** and **97.05%** Specificity. Hence, it is clear that **Random forest** is the better model in classifying the very low birth weight infant data.

5 As a second priority, **Bagging** under SOMTE has a high accuracy of **94.02 %** with sensitivity **90.90 %** and **97.05%** Specificity.

REFERENCES

- Margaret H Dunham (2005): Data Mining – Introductory and Advanced Topics, Pearson Education
- Rajan Chattamvelli (2009): Data Mining Methods, Narosa Publishing House
- R. Kumar and R. Varma (2012) "Classification algorithms for data mining: A survey," Int. J. Innov . Eng. Technol. IJIE
- N. Padhy (2012) "The Survey of Data Mining Applications and Feature Scope," Int. J. Compute . Sci. Eng. Inf. Technol
- Trevor Hastie, Robert Tibshirani and Jerome Friedman, The elements of statistical learning, Data mining, inference and prediction, Springer series
- Damodar N Guajarati (2003): basic Econometrics, McGraw Hill
- Classification of imbalanced data: areview (2001),Andrew Wong, Mohamed S.Kamel

Appendix

```
a=read.csv("Infa1.csv",header=T)
##MODE
mode=function(x){
ax=unique(x)
ax[which.max(tabulate(match(x,ax)))]
}
mode(race )
mode(twn)
mode(meth)
mode(delivery)
mode(vent)
mode(pneumo)
mode( pda)
mode(cld)
mode(pvh)
mode(ivh )
mode(ipe )
mode(dead)

#BARPLOT
par(mfrow=c(4,4))
barplot(table(race),main="race")
barplot(table(twn),main="twn")
barplot(table(meth),main="meth")
barplot(table(delivery),main="delivery")
barplot(table(vent),main="vent")
barplot(table(pneumo),main="pneumo")
barplot(table(pda),main="pda")
barplot(table(cld),main="cld")
barplot(table(pvh),main="pvh")
barplot(table(ivh),main="ivh")
barplot(table(ipe ),main="ipe ")
```

```
barplot(table(dead),main="dead")

#MEASURE OF CENTRAL TENDENCY
ds=x[c(1,2,3,4,5,7,8,13,21)];ds
mean=colMeans(ds);mean
vr=var(ds);vr
sd=sqrt(diag(vr));sd

##KURTOSIS AND SKEWNESS
k=kurtosis(ds)
sk=skewness(ds)
summary(ds)
#1 logistic regression
l=step(glm(dead~.,a,family="gaussian"))
summary(l)
###balanced data using smote method
library(ROSE)
library(smotefamily)
table(x$dead)
o11=ovun.sample(dead~.,x,method="over")
o1=o11$data
nrow(o1)
table(o1$dead)
write.csv(o1,file="birth11.csv")
X1=read.csv("birth11.csv",header=T)
head(X1)
length(X1)
X=X1[,-1];X
head(X)

###balanced data using rose method
library(ROSE)
library(smotefamily)
table(x$dead)
```

```
r1=ROSE(dead~ ., data =x, seed = 2)
r=r1$data
table(r$dead)
write.csv(r,file="birth12.csv")
X1=read.csv("birth12.csv",header=T);X1
head(X1)
length(X1)
X=X1[,-1];X
head(X)
length(X)
#splitting into test train
set.seed(1234)
library(caTools)
sp=sample.split(x,SplitRatio = 0.70)
tr=subset(x,sp == TRUE)
head(tr)
length(tr$dead)
trx=tr[,-10]
head(trx)
try=tr[,10]
ts=subset(x,sp==FALSE)
head(ts)
length(ts$dead)
tsx=ts[,-10]
head(tsx)
tsy=ts[,10]

### Imbalanced data
#1 logistic regression
g1=glm(dead~.,data=tr,family="binomial")
summary(g1)
p1=predict(g1,tsx,type = "response")
prob=as.data.frame(p1)
prob=round(prob,2)
```

```
p11=ifelse(prob>0.5,1,0)
library(caret)
t=table(tsy,p11)
ac1=sum(diag(t))/sum(t)*100
re1=t[1,1]/sum(t[1,])*100
pr1=t[2,2]/sum(t[2,])*100
c=confusionMatrix(as.factor(p11),as.factor(tsy))

#2 decision tree
library(rpart)
library(rpart.plot)
r=rpart(formula = dead~.,data=tr, method="class")
rpart.plot(r)
p2=predict(r,tsx,type="class")
library(caret)
t1=table(tsy,p2)
ac2=sum(diag(t1))/sum(t1)*100
re2=t1[1,1]/sum(t1[1,])*100
pr2=t1[2,2]/sum(t1[2,])*100
c1=confusionMatrix(as.factor(p2),as.factor(tsy))

#3 RandomForest
library(randomForest)
rf=randomForest(dead~.,data=tr)
p3=predict(rf,tsx)
p33=ifelse(p3>0.5,1,0)
library(caret)
t2=table(tsy,p33)
ac3=sum(diag(t2))/sum(t2)*100
re3=t2[1,1]/sum(t2[1,])*100
pr3=t2[2,2]/sum(t2[2,])*100
c2=confusionMatrix(as.factor(p33),as.factor(tsy))
```

```
#4 KNN
library(class)
k=knn(trx,tsx,try,k=3)
library(caret)
t3=table(tsy,k)
ac4=sum(diag(t3))/sum(t3)*100
re4=t3[1,1]/sum(t3[1,])*100
pr4=t3[2,2]/sum(t3[2,])*100
c3=confusionMatrix(as.factor(k),as.factor(tsy))

#5 SVM
library(e1071)
s=(tr$dead=as.factor(tr$dead))
sv=svm(dead~.,data=tr)
summary(sv)
p4=predict(sv,tsx)
library(caret)
t4=table(tsy,p4)
ac5=sum(diag(t4))/sum(t4)*100
re5=t4[1,1]/sum(t4[1,])*100
pr5=t4[2,2]/sum(t4[2,])*100
c4=confusionMatrix(as.factor(p4),as.factor(tsy))

# 6 Naive bayes
library(naivebayes)
nb=naive_bayes( dead~.,tr,usekernal=T)
p5=predict(nb,tsx,type="prob")
p55=predict(nb,tsx,type="class")
t5=table(tsy,p55)
ac6=sum(diag(t5))/sum(t5)*100
re6=t5[1,1]/sum(t5[1,])*100;re6
pr6=t5[2,2]/sum(t5[2,])*100
c5=confusionMatrix(as.factor(p55),as.factor(tsy))
```

```
#7 Linear Discriminant Analysis
library("MASS")
ld=lda(dead~.,data=tr)
summary(ld)
p7=predict(ld,tsx)$class
t7=table(tsy,p7)
ac8=sum(diag(t7))/sum(t7)*100
re8=t7[1,1]/sum(t7[1,])*100
pr8=t7[2,2]/sum(t7[2,])*100
c7=confusionMatrix(as.factor(p7),as.factor(tsy))

#8 Bagging
library(ipred)
cb=bagging(dead~.,data=tr)
p8=predict(cb,tsx)
t8=table(tsy,p8)
ac9=sum(diag(t8))/sum(t8)*100
re9=t8[1,1]/sum(t8[1,])*100
pr9=t8[2,2]/sum(t8[2,])*100
c8=confusionMatrix(as.factor(p8),as.factor(tsy))
```