STAFFING STABILITY PROGNOSIS

A Main Project submitted
in partial fulfilment of the requirements
for the award of the degree of

BACHELOR OF TECHNOLOGY
In
COMPUTER SCIENCE AND ENGINEERING

Submitted by

1. **P. Sony Priya (19PA1A05E9)**
2. **P. Bala Bharadwaj (19PA1A05F0)**
3. **Sk. Arshiya (20PA5A0517)**
4. **T. Teja Pavan Kumar (19PA1A05I0)**

Under the esteemed guidance of
M. Durga Satish
Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
VISHNU INSTITUTE OF TECHNOLOGY
(Autonomous)
(Approved by AICTE, Accredited by NBA & NAAC and permanently affiliated to JNTU Kakinada)
BHIMAVARAM – 534 202
2022 – 2023

VISHNU INSTITUTE OF TECHNOLOGY

(Autonomous)

(Approved by AICTE, Accredited by NBA & NAAC and permanently affiliated to JNTU Kakinada)

BHIMAVARAM-534202

2022-2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled “STAFFING STABILITY PROGNOSIS”, is being submitted by ***P. SONY PRIYA, P. BALA BHARADWAJ, SK. ARSHIYA AND T. TEJA PAVAN KUMAR***, bearing the **REGD.NOS: 19PA1A05E9, 19PA1A05F0, 20PA5A0517 and 19PA1A05I0** submitted in fulfilment for the award of the degree of **“BACHELOR OF TECHNOLOGY”** in **“COMPUTER SCIENCE AND ENGINEERING”** is a record of bonafide work carried out by them under my guidance and supervision during the academic year 2022-2023 and it has been found worthy of acceptance according to the requirements of university.

Internal Guide	**Head of the Department**
M. Durga Satish	Dr. Sumit Gupta

External Examiner

ACKNOWLEDGEMENT

It is nature and inevitable that the thoughts and ideas of other people tend to drift in to the subconscious due to various human parameters, where one feels acknowledge the help and guidance derived from others. We acknowledge each of those who have contributed for the fulfilment of this project.

We take the opportunity to express our sincere gratitude to **Dr D. Suryanarayana,** director and principal, VIT**,** Bhimavaram whose guidance from time to time helped us to complete this project successfully.

We are very much thankful to **Dr Sumit Gupta**, Head of the Department, Department of Computer Science and Engineering for his continuous and unrelenting support and guidance. We thank and acknowledge our gratitude to her for his valuable guidance and support expended to us right from the conception of the idea to the completion of this project.

We are very much thankful to **M. Durga Satish**, Assistant Professor, our internal guide whose guidance from time to time helped us to complete this project successfully.

Project Associates

P. Sony Priya	**(19PA1A05E9)**
P. Bala Bharadwaj	**(19PA1A05F0)**
Sk. Arshiya	**(20PA5A0517)**
T. Teja Pavan Kumar	**(19PA1A05I0)**

ABSTRACT

Employee turnover is a major concern for large companies as there is loss of valuable resource and extra resources like time and money has to be spent on recruiting new resources which directly affects their revenues. Due to this, companies are seeking means to predict factors that influence employee churn. The main goal of our work is to develop a churn prediction model that assists employers in predicting which employees are most likely to be subject to churn. The model developed in this work uses machine learning techniques to identify the underlying patterns so that they can be maneuvered to reduce churn. The model experimented with seven algorithms: Logistic regression, Random Forest, Bagging Classifier, KNN, Gradient Boost, Xgboost, and SVM.

Keywords: Logistic regression, Random Forest, Bagging Classifier, KNN, Gradient Boost, Xgboost, SVM.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.INTRODUCTION

Retaining an employee is one of the primary concerns to ensure company growth. An employee wanting to quit a job in a company or contract which causes undesirable consequences in organization is known as employee churn. Several bad experiences or even one – and an Employer may quit. And if herds of unsatisfied employees churn at once, there would be enormous damage to reputation and resources. Further discussion includes collecting data about client relationships with a brand and characteristics of employer behaviour that influence the turnover

The outcome of many research shows that the most valuable asset and important resource in organizations are their employees. Now a day due to increased competition and improved requirement in employees' proficiency determines the attrition rate. The employee attrition is considered to be a serious issue for organizations. The cost of searching and training employees is very high. Organizations need to search, hire and train new employees. Loss of experienced workers especially high performers is difficult to manage and is negatively related to the success and performance of organizations. The study focuses on the variables that may lead to control the attrition rate of the employee. The problem of employee turnover has turn to eminence in organizations because of its pessimistic impacts on issues on work place self-esteem and efficiency. The organizations deal with this problem is by predicting the risk of attrition of employees using machine learning techniques thus giving organizations to take proactive action for retention.

1.1 Problem definition

Employees opt for a product or a service for a particular period, which can be rather short–say, a month. Thus, an employee stays open for more interesting or advantageous offers. Plus, each time their current commitment ends, employees have a chance to reconsider and choose not to continue with the company. Of course, some natural churn is inevitable, and the figure differs from industry to industry. But having a higher churn figure than that is a definite sign that a business is doing something wrong. Even loyal employees won't tolerate it if they've had several issues with their company. For instance, in a survey by PricewaterhouseCoopers

(PwC), 59 percent of US employees responded that they will quit after multiple bad experiences, and 17 percent after just one bad experience.

1.2 Impact of Employee churn on business

Don't underestimate the impact of even a tiny percentage of churn, says Michael Redbird, general manager of Service Hub at HubSpot. "In a subscription-based business, even a small rate of monthly/quarterly churn will compound quickly over time. Just 1 percent monthly churn translates to almost 12 percent yearly churn. Given that it's far more expensive to acquire a new employer than to retain an existing one, businesses with high churn rates will quickly find themselves in a financial hole as they have to devote more and more resources to new employer acquisition.". According to this one survey by Invest, getting a new employer may cost up to five times more than retaining an existing employer. Churn rates directly affect revenue loss and acquisition spend rates. They also affect a company's growth potential, continues Michael, "Today's buyers aren't shy about sharing their experiences with vendors through channels like review sites and social media, as well as peer-to-peer networks. HubSpot Research found that 49 percent of buyers reported sharing an experience they had with a company on social media. In a world of eroding trust in businesses, word of mouth plays a more critical role in the buying process than ever before. From the same HubSpot Research study, 55 percent of buyers no longer trust the companies they buy from as much as they used to, 65 percent don't trust company press releases, 69 percent don't trust advertisements, and 71 percent don't trust sponsored ads on social networks."

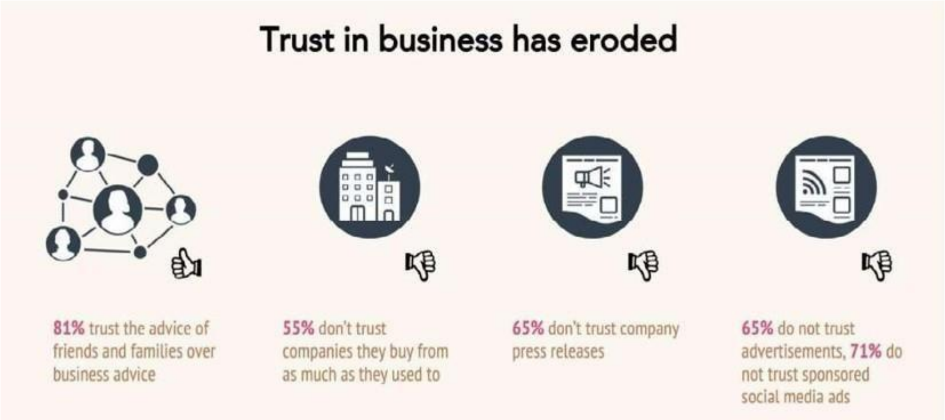


Figure 1.2 Trust in Business

1.3 Use case for employee churn

1.3.1 Telecom companies (cable or wireless)

Telecom companies that provide products and services like wireless network, internet, TV, cell phone (ATandT, Sprint, Verizon, Vodafone, etc.).

1.3.2 Video and music streaming services

Video and music streaming services are most commonly associated with subscription business model (Netflix, Amazon Video, Spotify etc.)

1.3.3 Software as a service provider

The adoption of cloud-hosted software is growing. According to Gartner, the SaaS market remains the largest segment of the cloud market. Its revenue is expected to grow 17.8 percent and reach.

1.3.4 Media

News companies offer readers digital subscriptions besides print ones (Bloomberg, The New York Times, Medium, etc.).

CHAPTER 2

PROBLEM STATEMENT

2.PROBLEM STATEMENT

Employee churn is a major concern for large companies as there is loss of valuable resource and extra resources like time and money has to be spent on recruiting new resources which directly affects their revenues. Due to this, companies are seeking means to predict factors that influence employee churn. The main goal of our work is to develop a churn prediction model that assists employers in predicting which employees are most likely to be subject to churn.

2.1 Limitations of Existing System

1.Present turnover predictors mainly focus on only customers but not on employees.

2.Most of the Existing systems use mainly logistic regression and KNN algorithms only.

3.It is very hard in the existing system to find out the churn rate as it requires a lot of time

4.Low Accuracy

2.2 Advantages of Proposed System

1.Identifying potential churners with different classification and clustering algorithms.

2.Testing different data sets with different classification algorithms.

3.Comparing the results of different classification algorithms.

4.High accuracy

2.3 Hardware and Software Requirements

2.3.1 Hardware Requirement:

1.Operating system: windows 10 or above

2.Ram: 4GB

3.Hard disk: 32 GB

4.Architecture:x86 64-bit CPU (Intel / AMD architecture) for python

5.Memory and disk space required per user: 1GB RAM + 1GB of disk +. 5 CPU core.

6.Server overhead: 2-4GB or 10% system overhead (whatever is larger). 5 CPU cores.

2.3.2 Software Requirements:

1.Python version above 3

2.Jupyter notebook/Google colab

3.Libraries required

- a. Pandas
- b. TensorFlow
- c. Scikit-learn
- d. Matplotlib
- e. Seaborn

2.4 Feasibility study

2.4.1 Economic Feasibility:

1. The cost of implementation is low.
2. As for the project, python and its libraries are used which are open-source (free to use).
3. For implementation, we require basic infrastructure.

2.4.2 Technical Feasibility:

For Staffing Stability Prognosis we are analyzing and training the model based on various parameters. The model experimented with seven algorithms: Logistic regression, Random Forest, Bagging Classifier, KNN, Gradient Boost, XGBoost, and SVM. As we are using python for building this Model we have so many open source libraries for analyzing the analytical data.

CHAPTER 3

DATASET DESCRIPTION

3.DATASET DESCRIPTION

The IBM Human Resource Analytic Employee Attrition and Performance dataset used in this project is a publicly available dataset from Kaggle Dataset Repository. It was IBM’s fictional dataset created by IBM data scientists. The dataset includes four (4) major components: employee satisfaction, income, seniority, and demographics data. The dataset contains several attributes influencing the predicted variable named ‘Attrition’ which signifies whether an employee left the company or not from 1,470 instances and 34 attributes. The identified class is labeled as ‘Attrition’ with 237 instances of ‘1’ and 1233 instances of ‘0’ having imbalanced data ratio of 1:5. The purpose of this study is to conduct a comparative study to develop machine learning models, i.e., KNN, Bagging Classifier, Logistic Regression, XgBoost, Random Forest, Gradient Boost and SVM for predicting probable employee attrition and compare between the algorithms in terms of their accuracy and efficiencies.

TABLE 3.1 Description of Attributes and Pre-Processing Action

No.	Feature Name	Type of Data	Data Description	Pre-processing action/Findings
1	Age	Continuous	The age of individual employee	Min = 18, max = 60 Normalize, Discretize
2	Attrition	Categorical	Employee leaving the company (Yes, No)	Set to class
3	BusinessTravel	Categorical	Business travel frequency (No Travel, Travel Frequently, Travel Rarely)	Retain
4	DailyRate	Continuous	Salary Level	Normalize, Discretize
5	Department	Nominal	Employee department (HR, R&D, Sales)	Retain
6	DistanceFromHome	Continuous	The distance from work to home	Min = 1, Max = 29 Normalize, Discretize
7	Education	Categorical	Level of education attained (1 ‘Below Collage’, 2= ‘College’, 3 = ‘Bachelor’, 4 = ‘Master’, 5 = ‘Doctor’)	Change to Nominal
8	EducationField	Nominal	Field of education (HR, Life Sciences, Marketing, Medical Sciences, Others, Technical)	Retain
9	EmployeeCount	Continuous	Count of instance	Cardinality = 1 - To remove
10	EmployeeNumber	Continuous	Employee ID	Cardinality = 1470 - To remove
11	EnvironmentSatisfaction	Categorical	Employee satisfaction with the environment (1 = 'Low', 2 = 'Medium', 3 = 'High', 4 = 'Very High')	Change to Nominal

12	Gender	Categorical	Female, Male)	Retain
13	HourlyRate	Continuous	Hourly Salary	Normalize, Discretize
14	JobInvolvem ent	Categorical	Job Involvement (1 = 'Low', 2 = 'Medium', 3 ='High', 4 = 'Very High')	Change to Nominal
15	JobLevel	Categorical	Level Of Job (1 to 5)	Change to Nominal
16	JobRole	Categorical	(1=HC Rep, 2=HR, 3=Lab Technician, 4=Manager, 5= Managing Director, 6=Research Director, 7= Research Scientist, 8=Sales Executive, 9= Sales Representative)	Retain
17	JobSatisfaction	Categorical	Satisfaction with the job (1= 'Low', 2 = 'Medium', 3 ='High', 4 = 'Very High')	Change to Nominal
18	MaritalStatus	Categorical	(1=Divorced, 2=Married, 3=Single)	Retain

19	MonthlyIncome	Continuous	Monthly Salary	Min = 1 009
				Max = 19 709
				Normalize, Discretize

20	MonthlyRate	Continuous	Monthly Rate	Normalize, Discretize

21	NumCompaniesWorked	Continuous	No. Of Companies Worked At	Min = 0
				Max = 9
				Normalize, Discretize

22	Over18	Categorical	(1=Yes, 2=No)	Cardinality = 1 To remove
23	OverTime	Categorical	(1=No, 2=Yes)	Retain
24	PercentSalaryHike	Continuous	Percentage Increase In Salary	Normalize, Discretize
25	PerformanceRating	Categorical	Performance Rating	Min = 3, Max = 4 Change to Nominal
26	RelationshipSatisfaction	Categorical	Relations Satisfaction (1 = 'Low', 2 = 'Medium', 3 = 'High', 4 = 'Very High')	Change to Nominal
27	StandardHours	Continuous	Standard Hours	Cardinality = 1 - To remove
28	StockOptionLevel	Categorical	Stock Options	Min = 0, Max = 3 Change to Nominal
29	TotalWorkin gYears	Continuous	Total Years Worked	Normalize, Discretize
30	TrainingTimesLastYear	Continuous	Hours Spent Training	Min = 0, Max = 6 Change to Nominal
31	WorkLifeBalance	Categorical	Time Spent Between Work and Outside (1 'Bad' 2 'Good' 3 'Better' 4 'Best')	Change to Nominal
32	YearsAtCom pany	Continuous	Total Number of Years at The Company	Min = 0, Max = 40 Normalize, Discretize
33	YearsInCurrentRole	Continuous	Years In Current Role	Min = 0, Max = 18 Normalize, Discretize
34	YearsSinceLastPromotion	Continuous	Last Promotion	Min = 0, Max = 15 Normalize, Discretize
35	YearsWithCurrManager	Continuous	Years Spent with Current Manager	Min = 0, Max = 17 Normalize, Discretize

CHAPTER 4

SYSTEM DESIGN

4. SYSTEM DESIGN

4.1 INTRODUCTION

System Design can be best understood from design goals and system architecture. Every system has its own system goals and significance.

4.1.1 DESIGN GOALS:

A few system design goals are as follows:

1. **Performance requirement:** All data entered shall be up to mark and no flaws shall be there for the performance to be 100%.

2. **Platform constraints:** The main target is to generate an intelligent system to predict the stability.

3. **Accuracy and Precision:** Requirements are the accuracy and precision of the data given as input as well as produced as output.

4. **Modifiability:** Requirements about the effort required to make changes in the software. (Person-months).

5. **Portability:** Since a mobile phone is handy so it is portable and can be carried and used whenever required.

6. **Reliability:** Requirements about how often the software fails. The definition of failure must be clear. Also, don't confuse reliability with availability which is quite a different kind of requirement. Be sure to specify the consequences of software failure, how to protect from failure, a strategy for error Prediction, and a strategy for correction.

7. **Security:** One or more requirements about the protection of your system and its data.

8. **Usability:** Requirements about how difficult it will be to learn and operate the system. The requirements are often expressed in learning time or similar

4.1.2 SYSTEM ARCHITECTURE

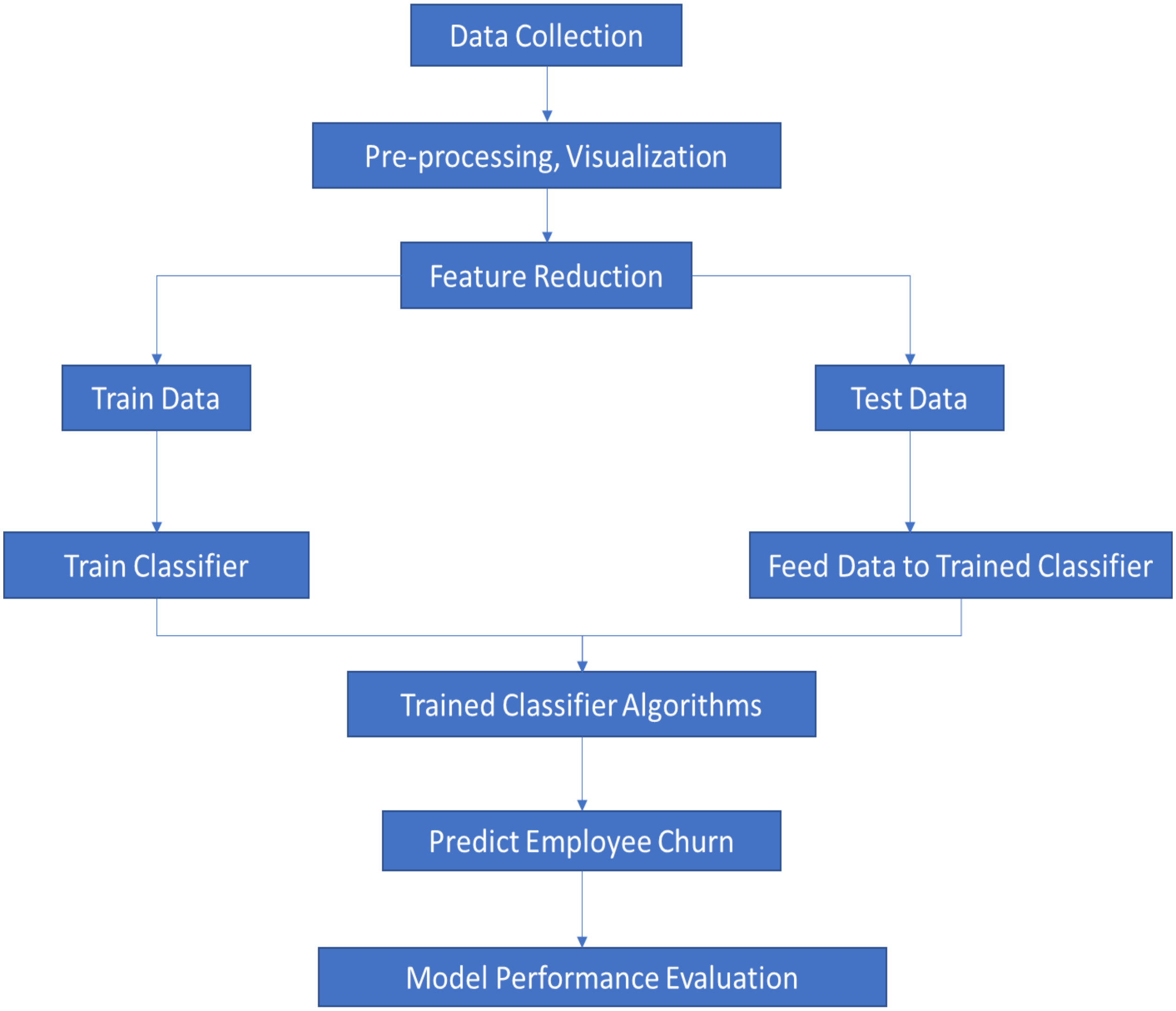

Data Collection
Pre-processing, Visualization
Feature Reduction
Train Data
Test Data
Train Classifier
Feed Data to Trained Classifier
Trained Classifier Algorithms
Predict Employee Churn
Model Performance Evaluation

Figure 4.1.2. System Architecture for Employee Turnover Prediction

4.1.3 Components of system architecture:

1.Data collection:

The first step is to gather data on company from. This could be in the form of a dataset that includes information such as the name of the employee, location, age, gender, salary etc.

2.Data cleaning and preparation:

Once the data has been collected, it is necessary to clean and prepare the data for analysis. This may involve removing any irrelevant or duplicate data, transforming data into a suitable format for analysis, and filling in missing values.

3.Data analysis:

The next step is to perform data analysis in Power BI. This may involve creating visualizations, such as bar charts and pie charts, to explore trends and patterns in the data. Additionally, statistical analysis may be used to determine relationships between different variables, such as the relationship between name of employee and salary.

4.Data interpretation:

The final step is to interpret the results of the analysis and draw meaningful conclusions.

5.Data presentation:

The results of the analysis should be presented in a clear and understandable manner, making use of the visualizations and data insights generated in Power BI.

4.2 UML DIAGRAMS

A model is an abstract representation of system, constructed to understand the system priority to building or modifying it. A model is a simplified representation of reality and it provides a means for conceptualization and communication of ideas in a precise and ambiguous form. We build models so that we can better understand the system we are developing. The elements are like components which can be associated in different ways to make a complete UML picture, which is known as diagram. Thus, it is very important to understand the different diagrams to implement the knowledge in real life systems.

UML (Unified Modelling Language) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems. It is a method for describing the system architecture in detail using the blueprint. We use UML diagrams to portray the behavior and structure of a system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed.

Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Once system requirements have been specified and analysed, system design is the first of the three technical activities - design, code and test that is required to build and verify software.

The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure, program structure, and procedural details are developed, reviewed, and documented. System design can be viewed from either a technical or project management perspective. From the technical point of view, design consists of four activities – architectural design, data structure You can also create your own set of diagrams to meet your requirements. Diagrams are generally made in an incremental and iterative way.

There are two broad categories of diagrams and they are again divided into subcategories –

1) Structural Diagrams
2) Behavioral Diagrams

4.2.1 Structural Diagrams:

The structural diagrams represent the static aspect of the system. These static aspects represent those parts of a diagram, which forms the main structure and are therefore stable. These static parts are represented by classes, interfaces, objects, components, and nodes. The four structural diagrams are –

1) Class diagram
2) Object diagram
3) Component diagram
4) Deployment diagram

4.2.2 Behavioral Diagrams:

Any system can have two aspects, static and dynamic. So, a model is considered as complete when both the aspects are fully covered. Behavioral diagrams basically capture the dynamic aspect of a system. Dynamic aspects can be further described as the changing/moving parts of a system. UML has the following five types of behavioral diagrams –

1) Use case diagram
2) Sequence diagram
3) Collaboration diagram
4) State chart diagram
5) Activity diagram

4.2.3 Use Case Diagram:

It represents the functionality of a system by utilizing actors and uses cases. It encapsulates the functional requirement of a system and its association with actors. It portrays the use case view of a system.

Following are the purposes of a use case diagram given below:

- It gathers the system's needs.
- It depicts the external view of the system.
- It recognizes the internal as well as external factors that influence the system.
- It represents the interaction between the actors.

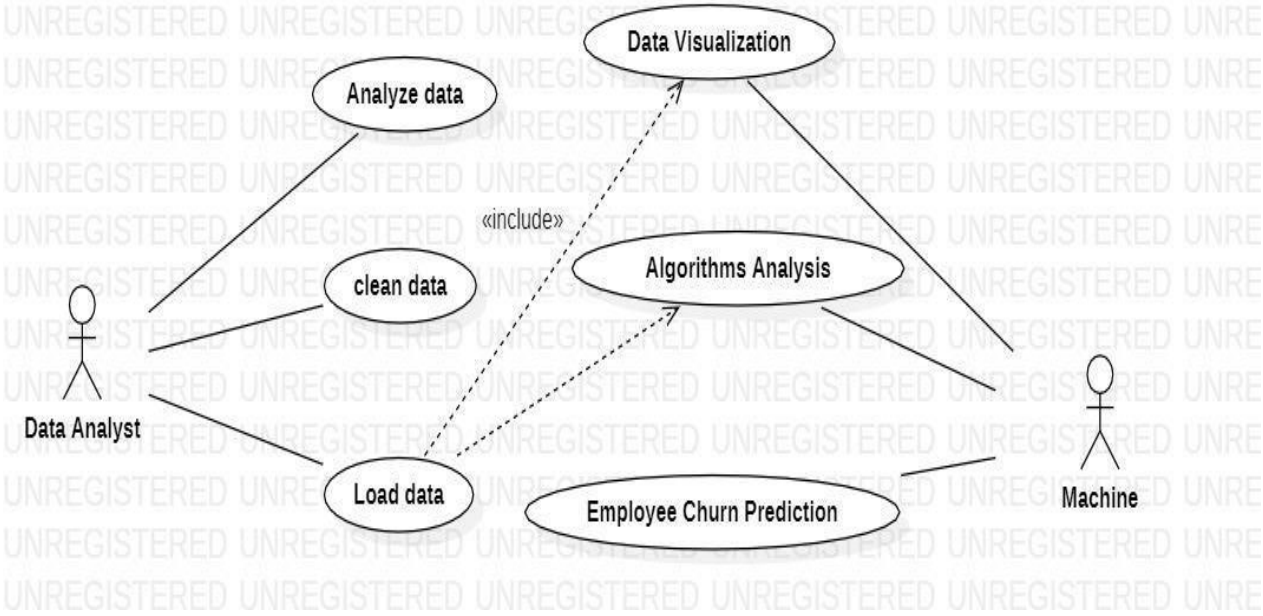


Figure 4.2.3 Use Case diagram for Employee Turnover Prediction

4.2.4 Sequence Diagram:

It shows the interactions between the objects in terms of messages exchanged over time. delineates in what order and how the object functions in a system.Following are the purposes of a sequence diagram given below:

- To model high-level interaction among active objects within a system.
- To model interaction among objects inside a collaboration realizing a use case.
- It either models' generic interactions or certain instances of interaction

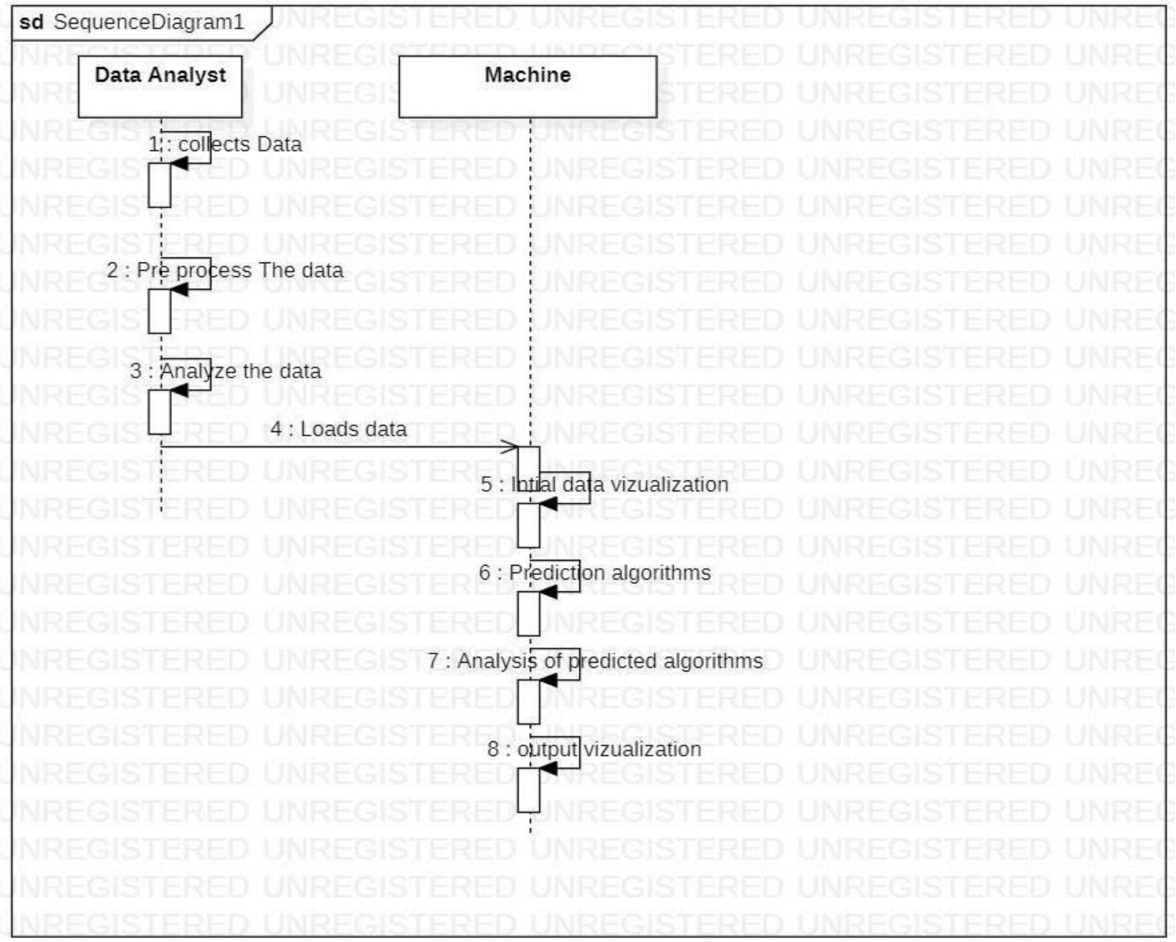


Figure 4.2.4 Sequence Diagram for Employee Turnover Prediction

4.2.5 Component Diagram:

Component diagrams represent a set of components and their relationships. These components consist of classes, interfaces, or collaborations. Component diagrams represent the implementation view of a system.

During the design phase, software artifacts (classes, interfaces, etc.) of a system are arranged in different groups depending upon their relationship. Now, these groups are known as components.

Finally, it can be said component diagrams are used to visualize the implementation

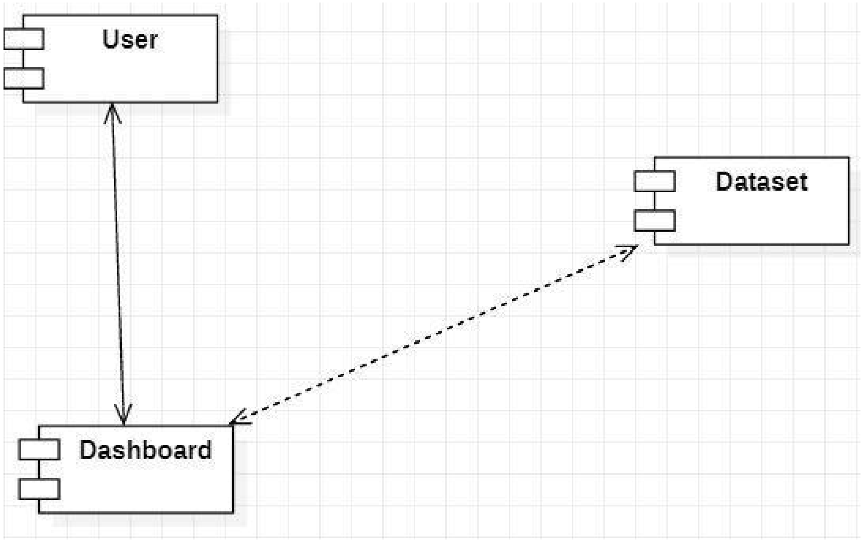


Figure 4.2.5 Component Diagram for Employee Turnover Prediction

CHAPTER 5

SYSTEM IMPLEMENTATION

5.SYSTEM IMPLEMENTATION

5.1 Data Gathering

- Hiring and retaining employees are extremely complex tasks that require capital, time and Skills.
- Small business owners spend 40% of their working hours on tasks that do not generate any income such as hiring.
- Companies spend 15%-20% of the employee's salary to recruit a new candidate.
- An average company loses anywhere between 1% and 2.5% of their total revenue on the time it takes to bring a new hire up to speed.
- Hiring a new employee costs an average of $7645 (0-500 corporation).
- It takes 52 days on average to fill a position.
- The HR team collected extensive data on their employees that could help us predict which employees are more likely to quit.
- The team provided you with an extensive data, here is a sample of the dataset:

1) Job Involvement
2) Education
3) Job Satisfaction
4) Performance Rating
5) Relationship Satisfaction
6) WorkLifeBalanceNumber of people who left based on gender

5.2 Data Pre-processing

1.Data Cleaning: In this step, we need to remove or impute missing values and outliers from the dataset. Missing values can be imputed using techniques such as mean or median imputation, or using more advanced methods such as K-nearest neighbors imputation or multiple imputation. Outliers can be detected using techniques such as boxplots and removed or imputed using appropriate methods.

2.Feature Selection: Feature selection is the process of selecting the most relevant features for the model. This step involves identifying the features that are most correlated with the target variable (employee churn) and removing redundant or irrelevant features. We can use correlation analysis, feature importance plots, or domain knowledge to identify the most relevant features.

3.Feature Scaling: In this step, we need to scale the features to a similar range to avoid any bias towards features with larger values. We can use techniques such as standardization or normalization to scale the features.

4.Data Splitting: In this step, we need to split the dataset into training and testing sets. The training set is used to train the model, and the testing set is used to evaluate the performance of the model on unseen data. We can use techniques such as random sampling, stratified sampling, or time-based splitting to split the dataset.

5.Feature Encoding: In this step, we need to convert categorical features into numerical features. We can use techniques such as one-hot encoding or label encoding to encode categorical features.

6.Data Balancing: In employee turnover prediction, the target variable may be imbalanced, meaning that there are more employees who did not churn than those who did. In this case, we need to balance the target variable using techniques such as oversampling or undersampling.

7.Model Training: After preprocessing the data, we can train the model using logistic regression, bagging classifier, XgBoost, KNN, Random forest, Gradient Boost and SVM algorithm.

These are the common data preprocessing steps involved in employee churn prediction using logistic regression, bagging classifier, , XgBoost, KNN, Random forest, Gradient Boost and SVM algorithm. However, the exact steps and techniques used may vary depending on the specific dataset and the requirements of the project.

5.2.1 Heatmap to show missing values - (none)

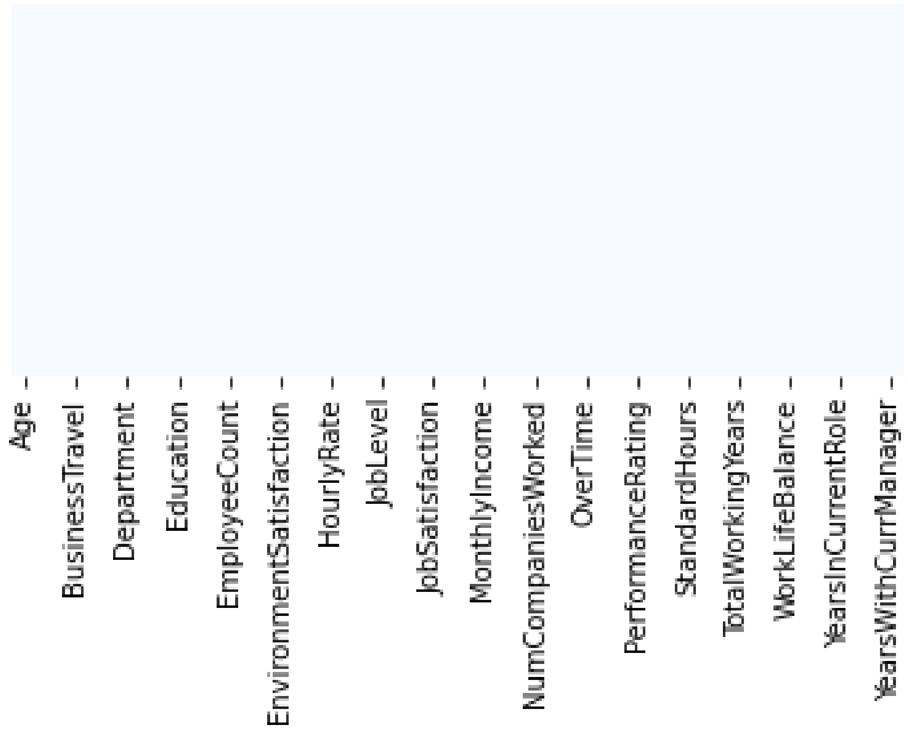


Figure 5.2.1 Heatmap

Several features such as ‘MonthlyIncome’ and ‘TotalWorkingYears’ are tail heavy.

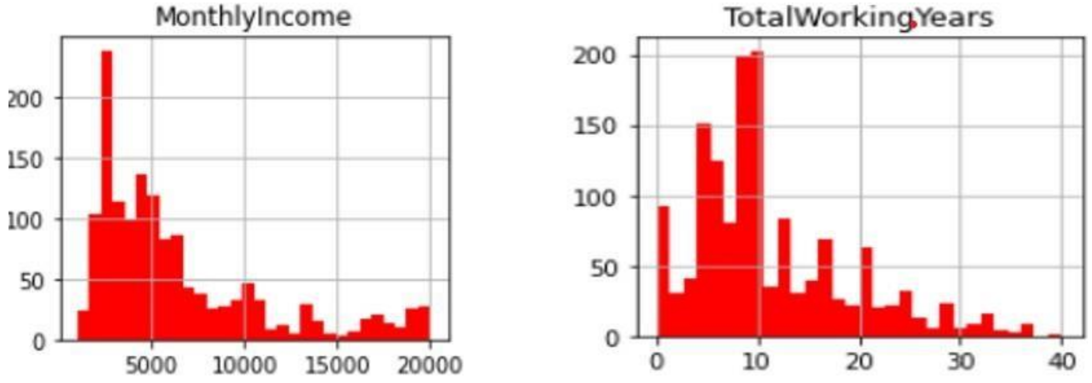


Figure 5.2.2 MonthlyIncome vs TotalWorkingYears

It makes sense to drop 'EmployeeCount'and 'Standardhours' and 'Over18' since they do not change from one employee to the other.

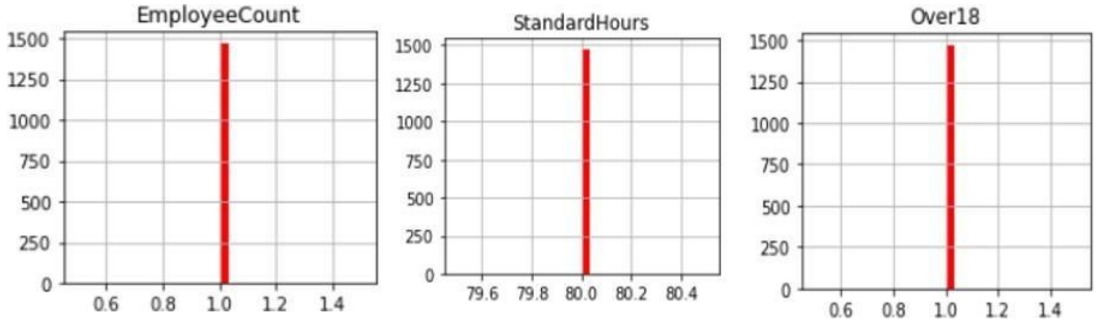


Figure 5.2.3 EmployeeCount vs StandardHours vs Over18

Total=1470

Number of employees who left the company = 237

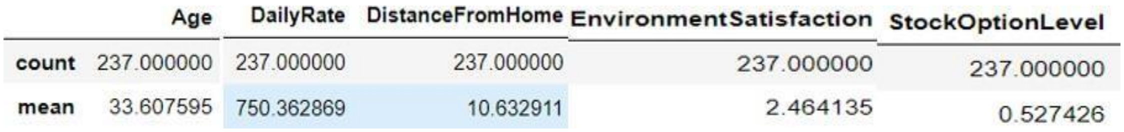
Percentage of employees who left the company = 16.122448979591837 %

Number of employees who did not left the company (stayed) = 1233

Percentage of employees who did not leave the company (stayed) = 83.87755102040816 %

5.2.4 Employees who stayed vs employees who left

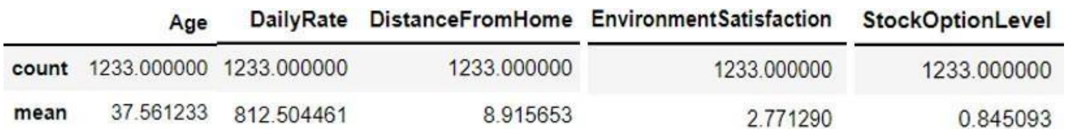
Left



	Age	DailyRate	DistanceFromHome	EnvironmentSatisfaction	StockOptionLevel
count	237.000000	237.000000	237.000000	237.000000	237.000000
mean	33.607595	750.362869	10.632911	2.464135	0.527426

Figure 5.2.4.1 Employees who left

Stayed



	Age	DailyRate	DistanceFromHome	EnvironmentSatisfaction	StockOptionLevel
count	1233.000000	1233.000000	1233.000000	1233.000000	1233.000000
mean	37.561233	812.504461	8.915653	2.771290	0.845093

Figure 5.2.4.2 Employees who stayed

5.3 Data Visualization

An overview to understand each attribute pattern should be carried out and examined through data visualization. From the data visualization, we can see that a few attributes need to be examined to ensure accuracy during the model classification process. Fig. 1 shows the data visualization of each attribute in the dataset.

5.3.1 Gender vs Attrition

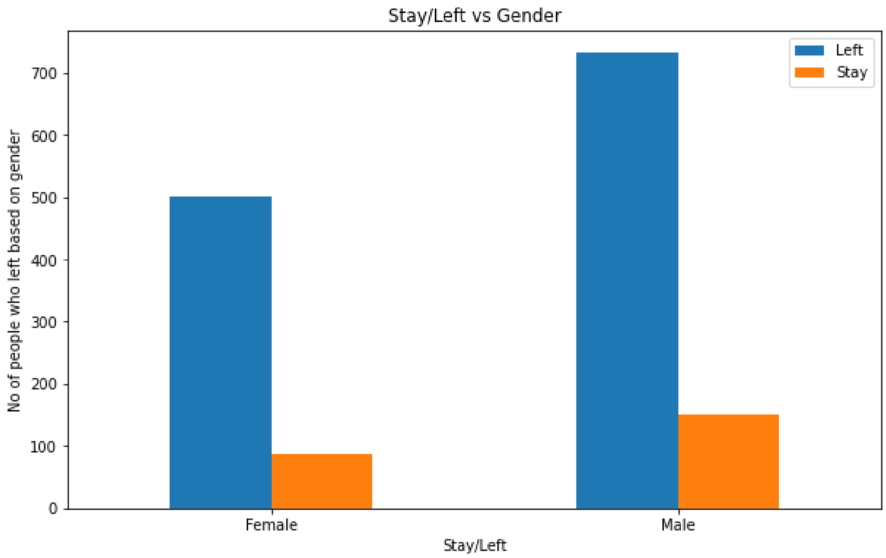


Figure 5.3 Gender vs Atrrition

Male employees tend to leave compared to female employees

5.3.2 Age vs Attrition

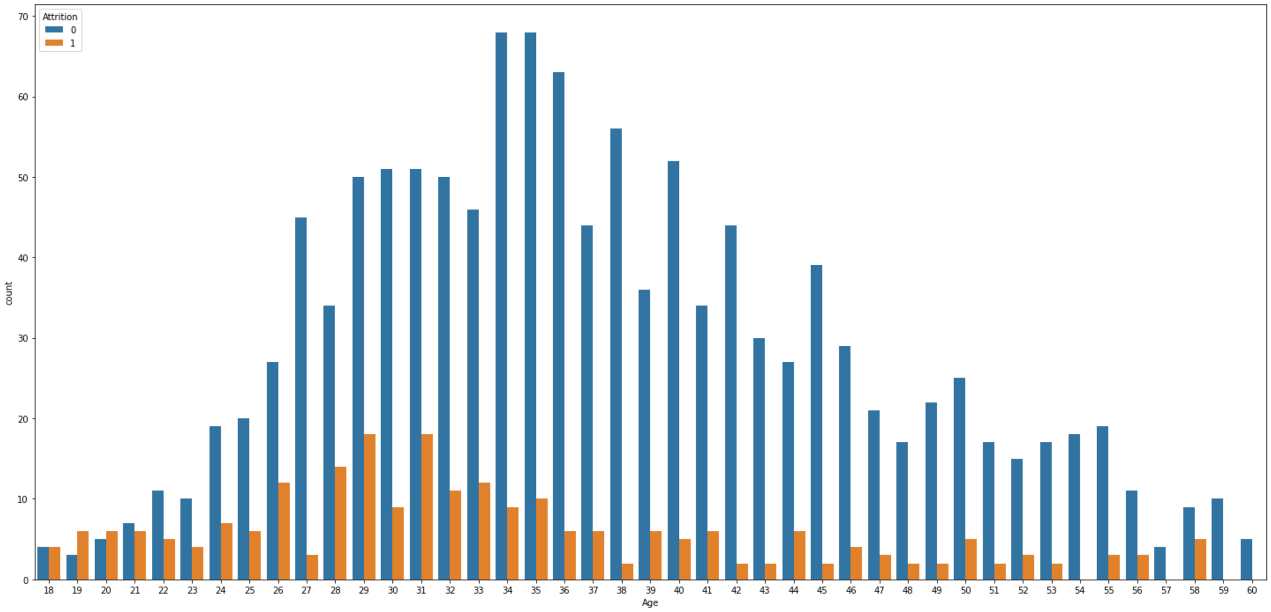


Figure 5.3.2 Age vs Attrition

Age 29-36 generate high frequency of people who stayed. Age 26-31 generate high frequency of people who left.

5.3.3 Business Travel vs Attrition

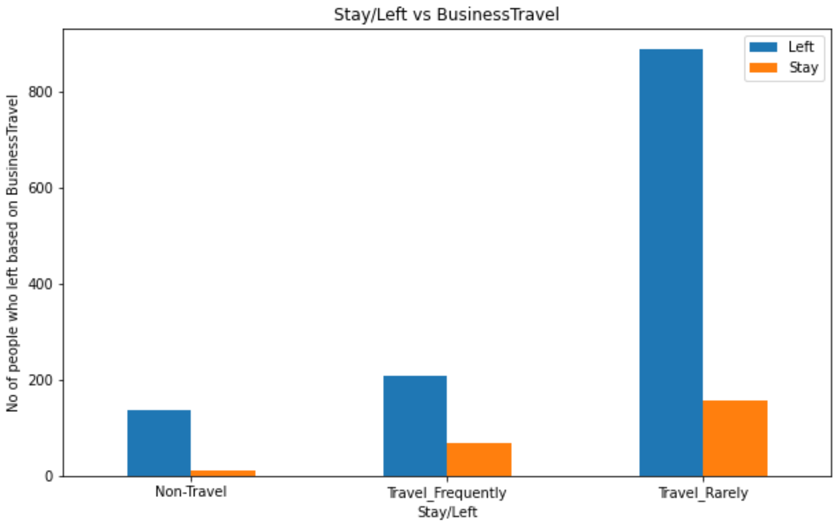


Figure 5.3.3 Business vs Attrition

Employees who travel rarely tend to leave compared to Non-Travel employees and employees who travel frequently.

5.3.4 Department vs Attrition

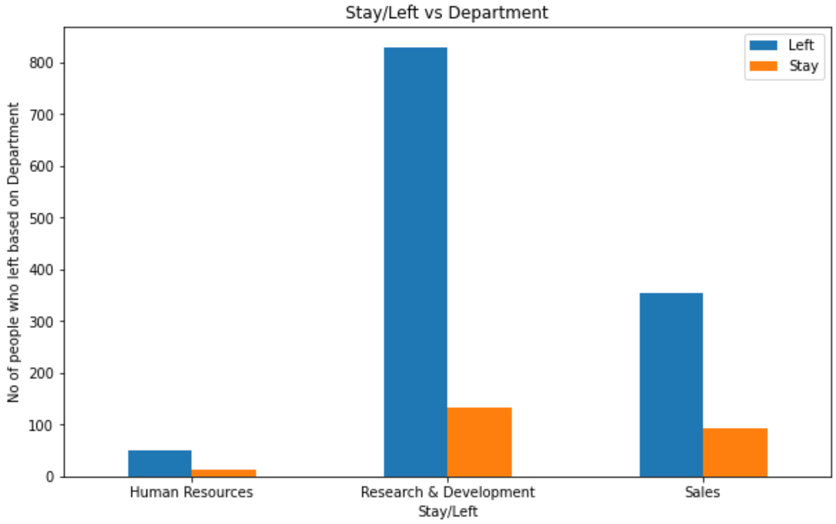


Figure 5.3.4 Department vs Attrition

Employees from Research and Development tend to leave compared to Human Resources and Sales.

CHAPTER 6

MODELS USED

6. MODELS USED

6.1 Logistic Regression

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1. Logistic Regression is much similar to the Linear Regression except that how they are used.

Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems. In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1). The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.

Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets. Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification.

6.1.1 Logistic Function (Sigmoid Function):

The sigmoid function is a mathematical function used to map the predicted values to probabilities. It maps any real value into another value within a range of 0 and 1. The value of the logistic regression must be between 0 and 1, which cannot go beyond this limit, so it forms a curve like the "S" form. The S-form curve is called the Sigmoid function or the logistic function. In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1. Such as values above the threshold value tends to 1, and a value below the threshold values tends to 0.

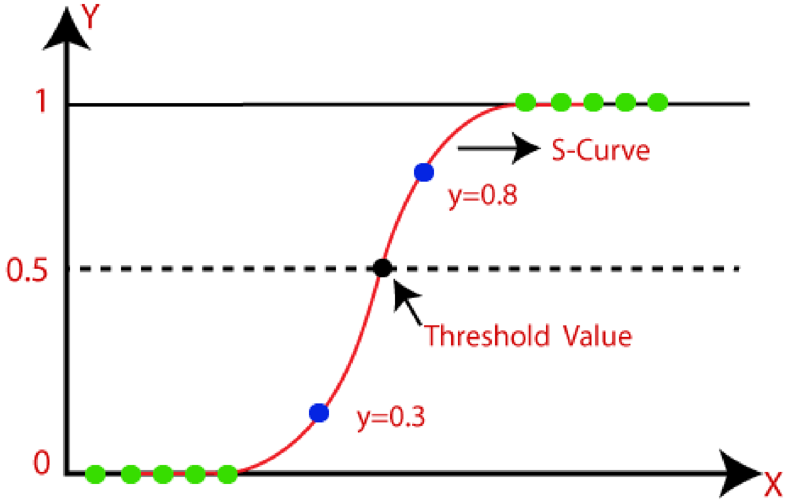


Figure 6.1.1: Logistic Function

6.2 Support Vector Machine

SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:

6.2.1 Hyperplane: There can be multiple lines/decision boundaries to segregate the classes in n-dimensional space, but we need to find out the best decision boundary that helps to classify the data points. This best boundary is known as the hyperplane of SVM.

The dimensions of the hyperplane depend on the features present in the dataset, which means if there are 2 features (as shown in image), then hyperplane will be a straight line. And if there are 3 features, then hyperplane will be a 2-dimension plane. We always create a hyperplane that has a maximum margin, which means the maximum distance between the data points.

6.2.2 Support Vectors: The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as Support Vector. Since these vectors support the hyperplane, hence called a Support vector.

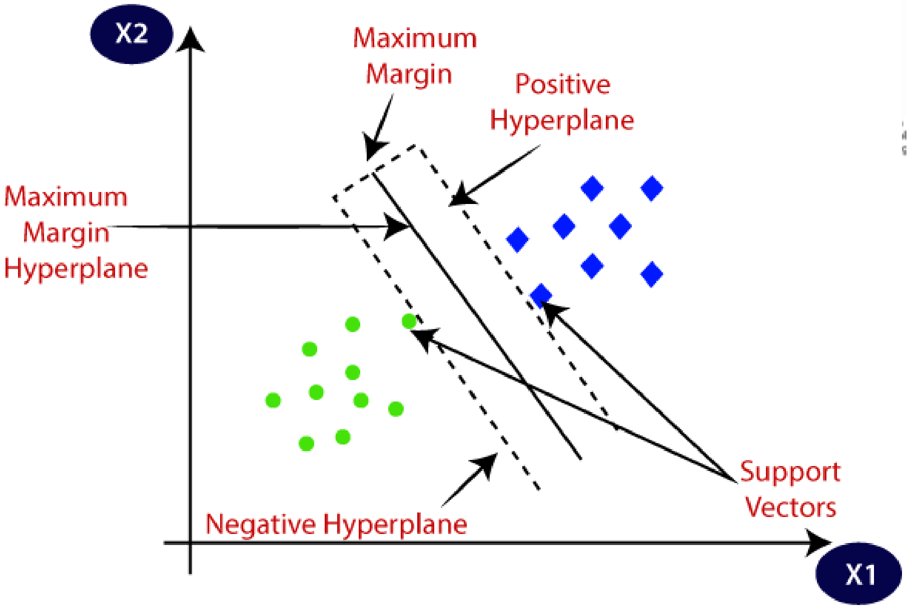


Figure 6.2.2 Hyperplane

6.2.3 Kernel Trick: A kernel is a method of placing a two dimensional plane into a higher dimensional space, so that it is curved in the higher dimensional space. (In simple terms, a kernel is a function from the low dimensional space into a higher dimensional space.)

However, for a non-linear data SVM finds it difficult to classify the data. The easy solution here is to use the Kernel Trick. A Kernel Trick is a simple method where a Non-Linear data is projected onto a higher dimension space so as to make it easier to classify the data where it could be linearly divided by a plane. This is mathematically achieved by Lagrangian formula using Lagrangian multipliers.

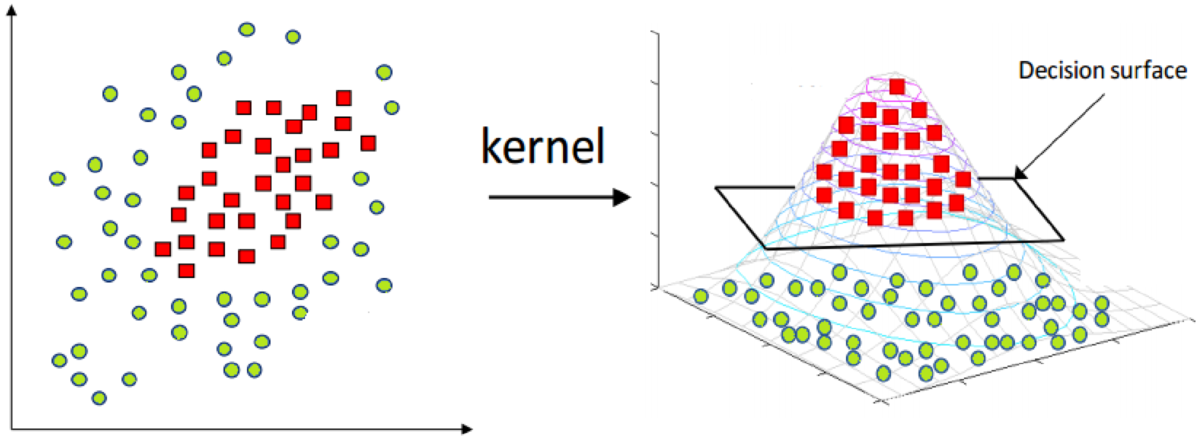


Figure 6.2.3 Kernal trick

6.3 K-Nearest Neighbor

K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique. K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset. KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

Example: Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.

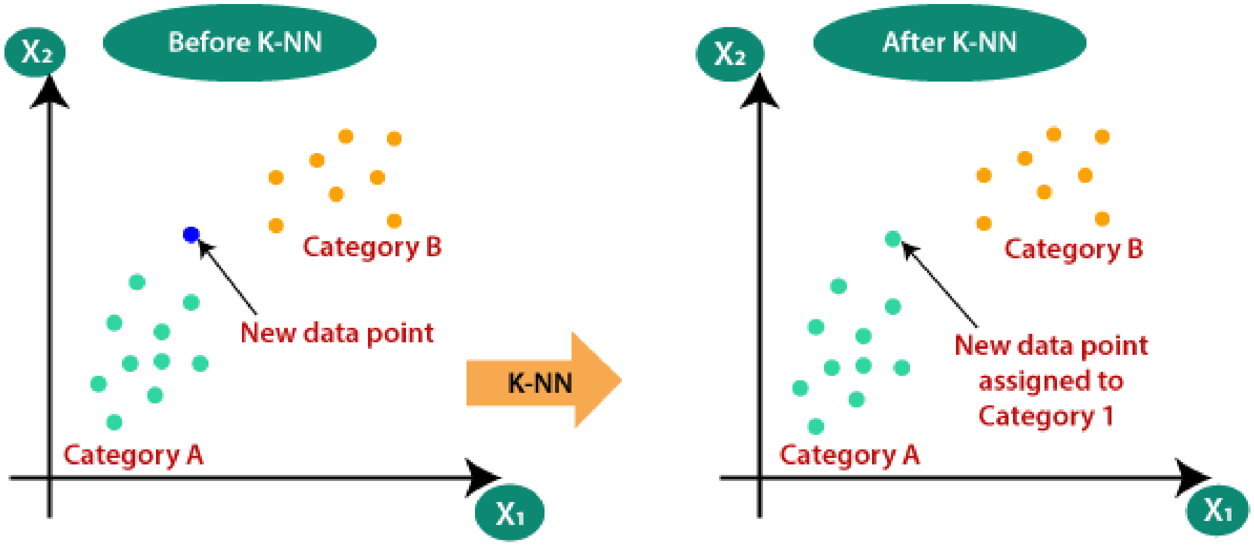


Figure 6.3 K-Nearest Neighbour

6.4 Bagging classifier

A Bagging classifier is an ensemble meta-estimator that fits base classifiers each on random subsets of the original dataset and then aggregate their individual predictions (either by voting or by averaging) to form a final prediction. Such a meta-estimator can typically be used as a way to reduce the variance of a black-box estimator (e.g., a decision tree), by introducing randomization into its construction procedure and then making an ensemble out of it. Each base classifier is trained in parallel with a training set which is generated by randomly drawing, with replacement, N examples from the original training dataset – where N is the size of the original training set. Training set for each of the base classifiers is independent of each other. Many of the original data may be repeated in the resulting training set while others may be left out.

6.4.1 How Bagging works on training dataset ?

How bagging works on an imaginary training dataset is shown below. Since Bagging resamples the original training dataset with replacement, some instance (or data) may be present multiple times while others are left out.

```
Classifier generation:
        Let N be the size of the training set.
        for each of t iterations:
            sample N instances with replacement from the original training set.
            apply the learning algorithm to the sample.
            store the resulting classifier.

        Classification:
        for each of the t classifiers:
            predict class of instance using classifier.
        return class that was predicted most often.
```

Bagging is one of the Ensemble construction techniques, which is also known as Bootstrap Aggregation. Bootstrap establishes the foundation of the Bagging technique. Bootstrap is a sampling technique in which we select "n" observations from a population of "n" observations. But the selection is entirely random, i.e., each observation can be chosen from the original population so that each observation is equally likely to be selected in each iteration of the bootstrapping process. After the bootstrapped samples are formed, separate models are trained with the bootstrapped samples. In real experiments, the bootstrapped samples are drawn from the training set, and the sub-models are tested using the testing set. The final output prediction is combined across the projections of all the sub-models.

6.4 Random Forest Classifier

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.The below diagram explains the working of the Random Forest algorithm:

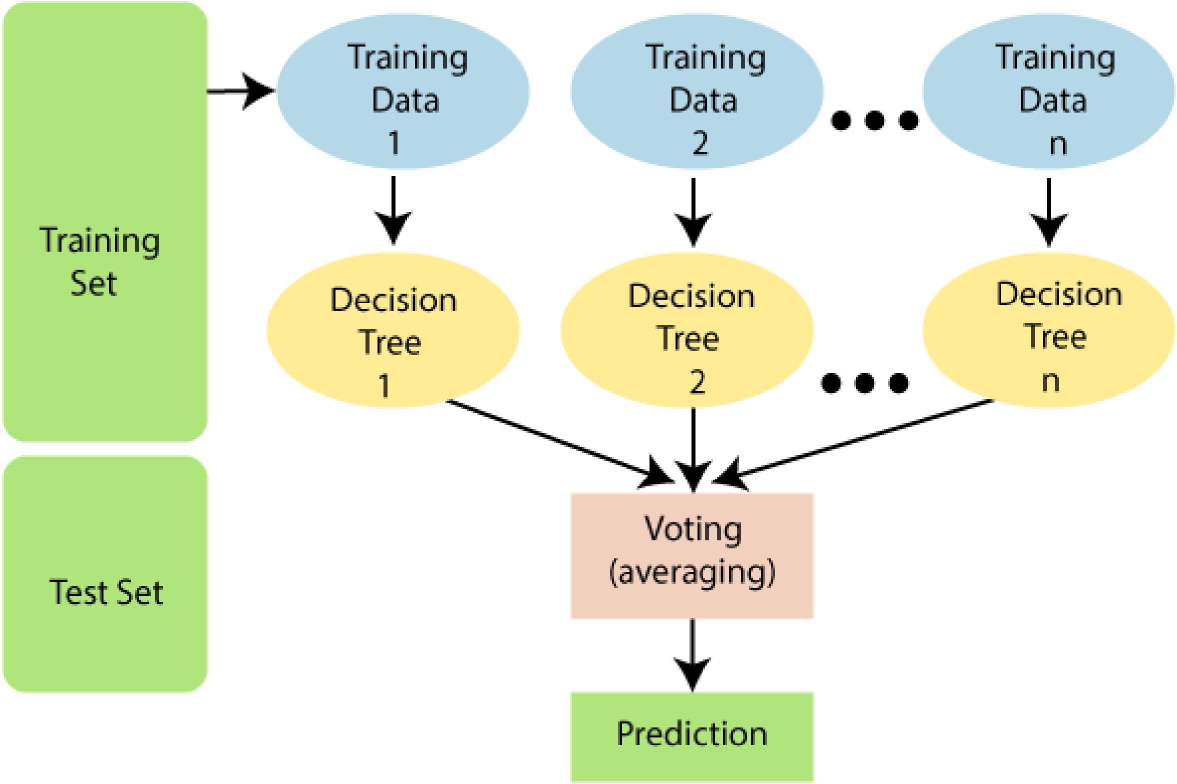


Figure 6.4 Random Forest Classifier

6.5 Gradient Boosting Classifier

Gradient Boosting Machine (GBM) is one of the most popular forward learning ensemble methods in machine learning. It is a powerful technique for building predictive models for regression and classification tasks. GBM helps us to get a predictive model in form of an ensemble of weak prediction models such as decision trees. Whenever a decision tree performs as a weak learner then the resulting algorithm is called gradient-boosted trees.

When the target column is continuous, we use Gradient Boosting Regressor whereas when it is a classification problem, we use Gradient Boosting Classifier. The only difference between the two is the “Loss function”. The objective here is to minimize this loss function by adding weak learners using gradient descent. Since it is based on loss function hence for regression problems, we'll have different loss functions like Mean squared error (MSE) and for classification, we will have different for e.g log-likelihood.

It enables us to combine the predictions from various learner models and build a final predictive model having the correct prediction. But here one question may arise if we are applying the same algorithm then how multiple decision trees can give better predictions than a single decision tree? Moreover, how does each decision tree capture different information from the same data?

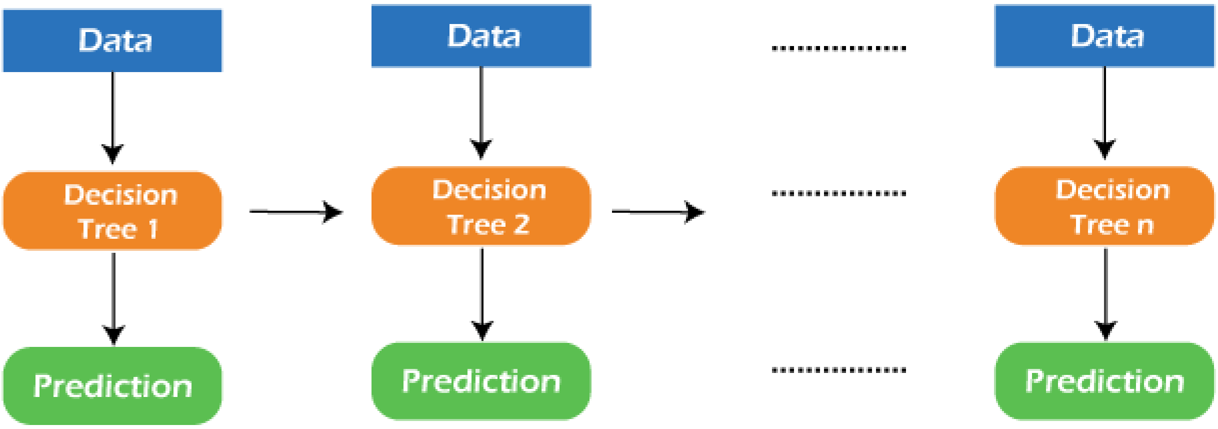


Figure 6.5 Gradient Boosting Classifier

So, the answer to these questions is that a different subset of features is taken by the nodes of each decision tree to select the best split. It means, that each tree behaves differently, and hence captures different signals from the same data.

6.6 XgBoost Classifier

XGBM is the latest version of gradient boosting machines which also works very similar to GBM. In XGBM, trees are added sequentially (one at a time) that learn from the errors of previous trees and improve them. Although, XGBM and GBM algorithms are similar in look and feel but still there are a few differences between them as follows:

XGBM uses various regularization techniques to reduce under-fitting or over-fitting of the model which also increases model performance more than gradient boosting machines.

XGBM follows parallel processing of each node, while GBM does not which makes it more rapid than gradient boosting machines.

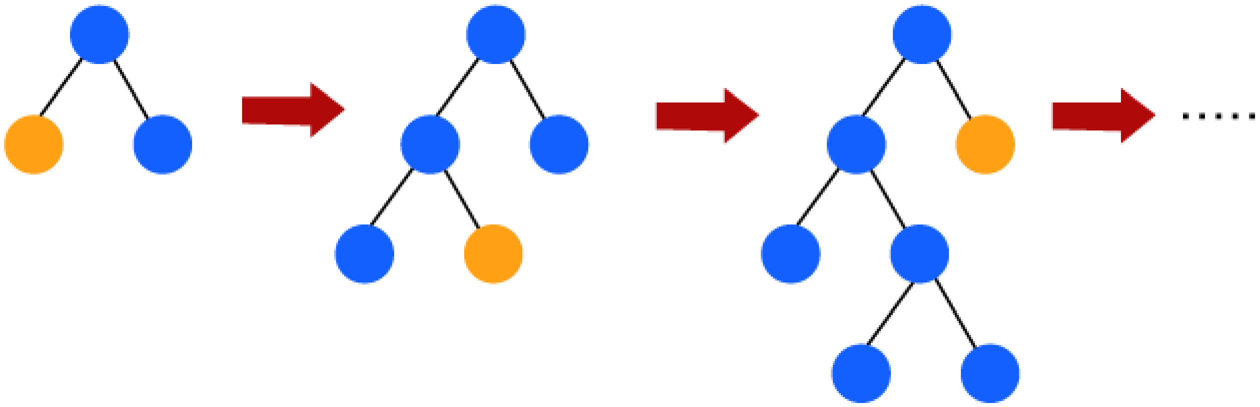


Figure 6.6 XGBoost Classifier

XGBM helps us to get rid of the imputation of missing values because by default the model takes care of it. It learns on its own whether these values should be in the right or left node.

CHAPTER 7

EXPERIMENTAL SETUP

7.EXPERIMENTAL SETUP

7.1 How training data is given (Train test, cross validation etc):

7.1.1 Splitting the Data Set:

Splitting the dataset is one of the major parts of building a Machine learning model. Generally, for Machine learning problems, we split the dataset into two parts - the Training sets and Test sets. But for Deep learning problems, we split the dataset into three parts - Training data, Validation data, and Test dataset. Take a look at the picture below:

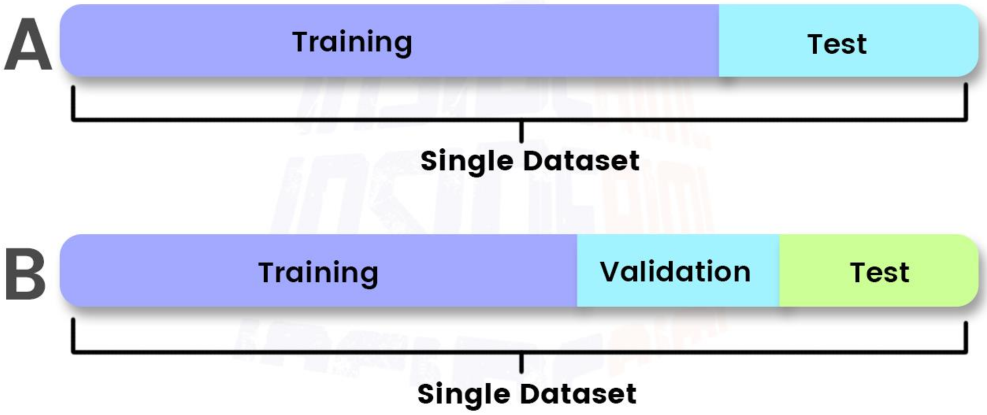


Figure 7.1.1 Splitting the Dataset

7.1.2 Training Set

This is the actual dataset from which a model trains i.e., the model sees and learns from this data to predict the outcome or to make the right decisions. Most of the training data is collected from several resources and then Pre-processed and organized to provide proper performance of the model. Type of training data hugely determines the ability of the model to generalize. i.e., the better the quality and diversity of training data, the better will be the performance of the model. This data is more than 60% of the total data available for the project.

7.1.3 Testing Set

This dataset is independent of the training set but has a somewhat similar type of probability distribution of classes and is used as a benchmark to evaluate the model, used only after the training of the model is complete. Testing set is usually a properly organized dataset having all kinds of data for scenarios that the model would probably be facing when used in the real world.

Often the validation and testing set combined is used as a testing set which is not considered a good practice. If the accuracy of the model on training data is greater than that on testing data then the model is said to have overfitting. This data is approximately 20-25% of the total data available for the project.

7.1.4 Validation Set

The validation set is used to fine-tune the hyperparameters of the model and is considered a part of the training of the model. The model only sees this data for evaluation but does not learn from this data, providing an objective unbiased evaluation of the model. Validation dataset can be utilized for regression as well by interrupting training of model when loss of validation dataset becomes greater than loss of training dataset. i.e., reducing bias and variance. This data is approximately 10-15% of the total data available for the project but this can change depending upon the number of hyperparameters. i.e. if model has quite many hyperparameters then using large validation set will give better results. Now, whenever the accuracy of model on validation data is greater than that on training data then the model is said to have generalized well.

7.1.5 Preparing the training and test data:

Our data has 1470 records which consists of employee details. We are going to use 80% of the data for training the model and 20% of the data for evaluating the model. X contains features of the dataset and Y contains our target variable or the column which we need to predict i.e., ‘Attrition’.

7.1.6 How Training Data is given:

That training data and the test data are taken from the human resources dataset from Kaggle by splitting it in the ratio 80:20 by using the method train_test_split in sklearn.model_selection where Sklearn is a library in python.

7.1.7 Train_Test_Split:

It is a function in Sklearn model selection of splitting data arrays into two subsets: for training data and for testing data. With this function, you don't need to divide the dataset manually. By default, Sklearn train_test_split will make random partitions for the two subsets. However, you can also specify a random state for the operation.

7.2 Learning rates and Parameters:

In machine learning and statistics, the learning rate is a turning parameter in an optimization algorithm that determines the step size at each iteration while moving toward a minimum of a loss function. Since it influences to what extent newly acquired information overrides old information, it metaphorically represents the speed at which machine learning model learns.

7.2.1 Random_State:

It is used for initializing the internal random number generator, which will decide the splitting of data into train and test indices in your case. In the documentation, it is stated that: If random_state is None or np. random, then a randomly initialized Random State object is returned.

Parameters:

X_train: The feature train set.

Y_train: The training target values.

X_test: The set used for training purpose.

Y_test: The classification values of the test set of data.

CHAPTER 8

EVALUATING MODEL

8.EVALUATING MODEL

Model Evaluation is an integral part of the model development process. It helps to find the best model that represents our data and how well the chosen model will work in the future. Evaluating model performance with the data used for training is not acceptable in data science because it can easily generate overoptimistic and overfitted models.

8.1 Evaluation Metrics:

Model evaluation metrics are required to quantify model performance. The choice of evaluation metrics depends on a given machine learning task (such as classification, regression, ranking, clustering, topic modelling, among others). Some metrics, such as precision-recall, are useful for multiple tasks. Supervised learning tasks such as classification and regression constitute a majority of machine learning applications.

8.1.1 Classification Accuracy:

Accuracy is one metric for evaluating classification models. Informally, **accuracy** is the fraction of predictions our model got right. Formally, accuracy has the following definition:

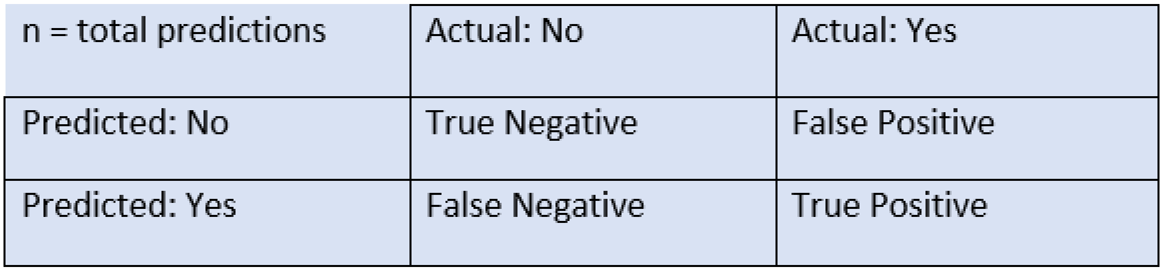
Accuracy=Number of correct predictions/Total number of predictions

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

8.1.2 Confutsion Matrix:

The confusion matrix is a matrix used to determine the performance of the classification models for a given set of test data. It can only be determined if the true values for test data are known. The matrix itself can be easily understood, but the related terminologies may be confusing. Since it shows the errors in the model performance in the form of a matrix, hence also known as an **error matrix**. Some features of Confusion matrix are given below:

- For the 2 prediction classes of classifiers, the matrix is of 2*2 table, for 3 classes, it is 3*3 table, and so on.
- The matrix is divided into two dimensions, that are **predicted values** and **actual values** along with the total number of predictions.
- Predicted values are those values, which are predicted by the model, and actual values are the true values for the given observations.
- It looks like the below table:



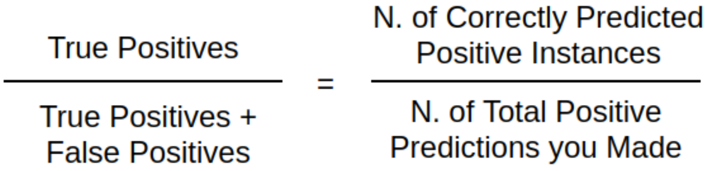
n = total predictions	Actual: No	Actual: Yes
Predicted: No	True Negative	False Positive
Predicted: Yes	False Negative	True Positive

The above table has the following cases:

- **True Negative:** Model has given prediction No, and the real or actual value was also No.
- **True Positive:** The model has predicted yes, and the actual value was also true.
- **False Negative:** The model has predicted no, but the actual value was Yes, it is also called as **Type-II error**.
- **False Positive:** The model has predicted Yes, but the actual value was No. It is also called a **Type-I error.**

Precision

Precision is a measure of how many of the positive predictions made are correct (true positives). The formula for it is:



$$\frac{\text{True Positives}}{\text{True Positives + False Positives}} = \frac{\text{N. of Correctly Predicted Positive Instances}}{\text{N. of Total Positive Predictions you Made}}$$

Recall / Sensitivity

Recall is a measure of how many of the positive cases the classifier correctly predicted, over all the positive cases in the data. It is sometimes also referred to as Sensitivity. The formula for it is:

$$\frac{\text{True Positives}}{\text{True Positives + False Negatives}} = \frac{\text{N. of Correctly Predicted Positive Instances}}{\text{N. of Total Positive Instances in the Dataset}}$$

Specificity

Specificity is a measure of how many negative predictions made are correct (true negatives). The formula for it is:

$$\frac{\text{True Negatives}}{\text{True Negatives + False Positives}} = \frac{\text{N. of Correctly Predicted Negative Instances}}{\text{N. of Total Negative Instances in the Dataset}}$$

F1-Score

F1-Score is a measure combining both precision and recall. It is generally described as the harmonic mean of the two. Harmonic mean is just another way to calculate an “average” of values, generally described as more suitable for ratios (such as precision and recall) than the traditional arithmetic mean. The formula used for F1-score in this case is:

$$2 * \frac{\text{Precision * Recall}}{\text{Precision + Recall}}$$

8.2 Logistic Regression:

One of the simplest algorithms in machine learning is Logistic Regression. It is employed to address classification issues. The likelihood of an event is determined mathematically using a sigmoid function. Then the observation is put into its respective class. When calculating, a threshold value is chosen, and classes with probabilities above the threshold are given the value 1, and classes with probabilities below the threshold are given the value 0.

Accuracy 83.42391304347827 %

	Precision	Recall	F1-Score	Support
0	0.84	1	0.91	305
1	0	0	0	60
Accuracy			0.84	368
Macro Avg	0.42	0.5	0.46	368
Weighted Avg	0.70	0.84	0.76	368

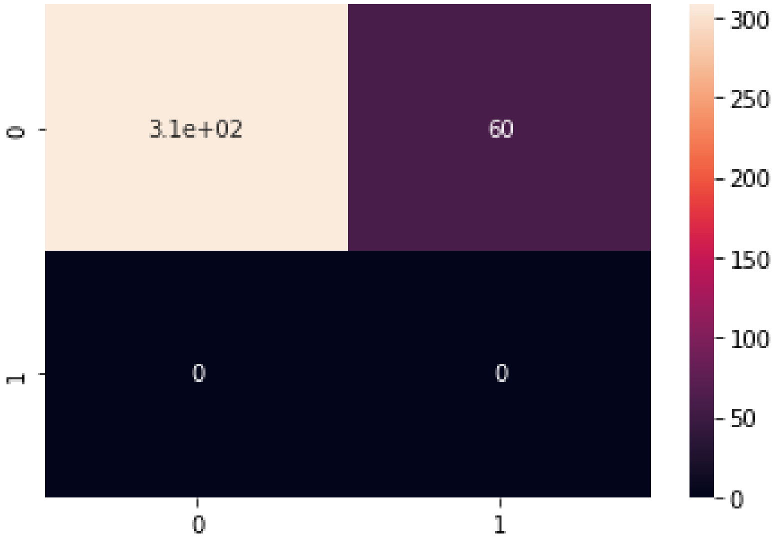


Table 8.2 Confusion Matrix for Logistic Regression

8.3 K-Nearest Neighbor:

The algorithm starts by storing all of the training data. When a new data point is encountered, its distance from all of the stored points is calculated. The K nearest neighbors to the new data point are selected based on the distances calculated in the previous step. The value of K is a user-defined parameter, usually set to an odd number to avoid ties in the voting process. The class label or target value is assigned to the new data point based on either the majority class of the K nearest neighbors (for classification problems) or the average value of the K nearest neighbors (for regression problems).

Accuracy 83.69565217391305 %

	Precision	Recall	F1-Score	Support
0	0.84	1	0.91	305
1	0	0	0	60
Accuracy			0.84	368
Macro Avg	0.42	0.5	0.46	368
Weighted Avg	0.70	0.84	0.76	368

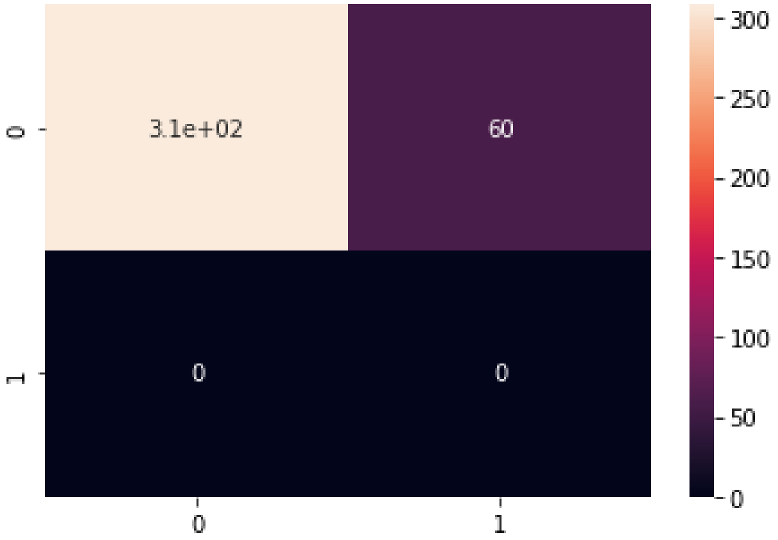


Table 8.3 Confusion Matrix for KNN

8.4 Support Vector Machine:

Each data point in the dataset is plotted in SVM in an N-dimensional space, where N is the total number of features and attributes in the data. Next, choose the best hyperplane for separating the data. As a result, you must now realize that SVM can only conduct binary classification by nature (i.e., choose between two classes). For multi-class problems, there are numerous approaches to apply. Multi-Class Support Vector Machine for Problems We can develop a binary classifier for each category of the data in order to execute SVM on multi-class problems. Each classifier's two outcomes will be: The data point is either a member of that class OR it is not a member of that class.

Accuracy 83.69565217391305 %

	Precision	Recall	F1-Score	Support
0	0.84	1	0.91	305
1	0	0	0	60
Accuracy			0.84	368
Macro Avg	0.42	0.5	0.46	368
Weighted Avg	0.70	0.84	0.76	368

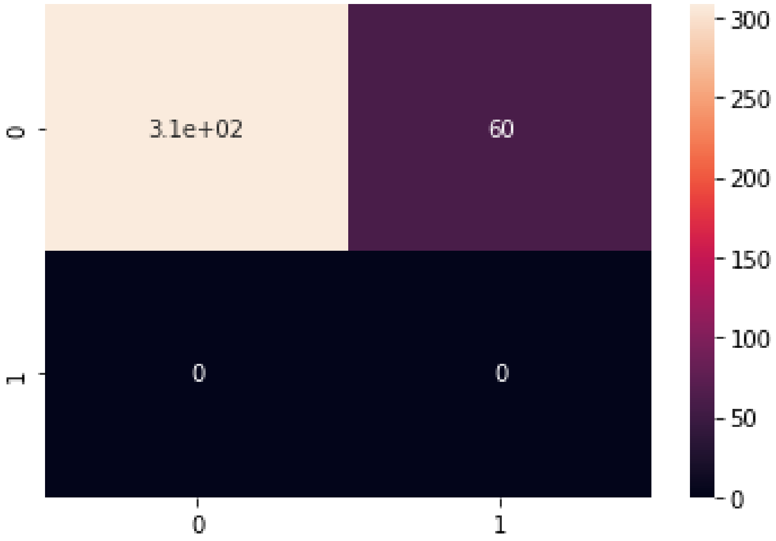


Table 8.4 Confusion Matrix for SVM

8.5 Bagging Classifier:

A Bagging classifier is an ensemble meta-estimator that fits base classifiers each on random subsets of the original dataset and then aggregates their individual predictions to form a final prediction. Such a meta-estimator can typically be used as a way to reduce the variance of a black-box estimator, by introducing randomization into its construction procedure and then making an ensemble out of it. Each base classifier is trained in parallel with a training set which is generated by randomly drawing, with replacement, examples (or data) from the original training dataset. The training set for each of the base classifiers is independent of each other. Many of the original data may be repeated in the resulting training set while others may be left out. Bagging reduces overfitting (variance) by averaging or voting, however, this leads to an increase in bias, which is compensated by the reduction in variance though.

Accuracy 83.69565217391305 %

	Precision	Recall	F1-Score	Support
0	0.84	1	0.91	305
1	0	0	0	60
Accuracy			0.84	368
Macro Avg	0.42	0.5	0.46	368
Weighted Avg	0.70	0.84	0.76	368

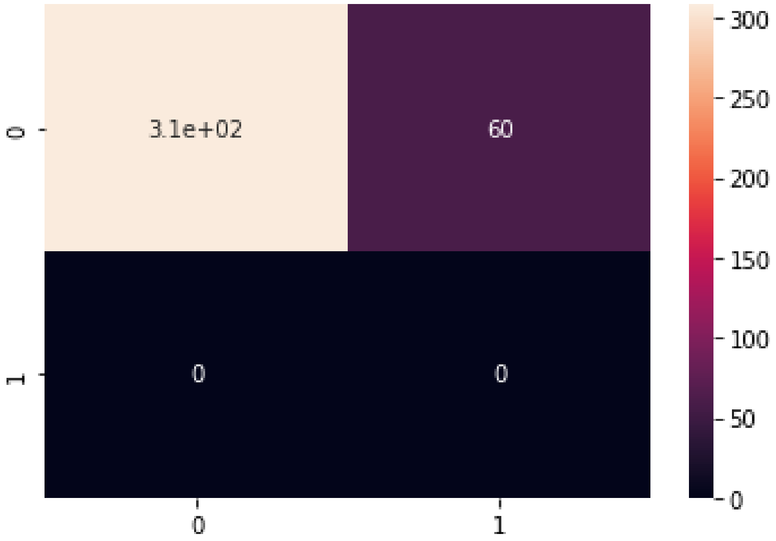


Table 8.5 Confusion Matrix for Bagging Classifier

8.6 Random Forest Classifier:

The ML algorithm Random Forest is used in this project. The output will be divided based on the number of decision trees created in the training scenario. classification, classification prediction, or regression. Prediction accuracy is inversely correlated with the number of trees. Rainfall, perception, temperature, and productivity are all included in the dataset. These dataset factors are used for training. Only two-thirds of the dataset is taken into account. The remaining dataset is used as the foundation for experiments. Three parameters for the random forest method include: The terms "n tree" and "m tree" refer to the number of variables that must be considered when splitting a node. The number of nodes in the terminal nodes tells us how many observations are required.

Accuracy 85.48565217391305 %

	Precision	Recall	F1-Score	Support
0	0.84	1	0.91	305
1	0	0	0	60
Accuracy			0.84	368
Macro Avg	0.42	0.5	0.46	368
Weighted Avg	0.70	0.84	0.76	368

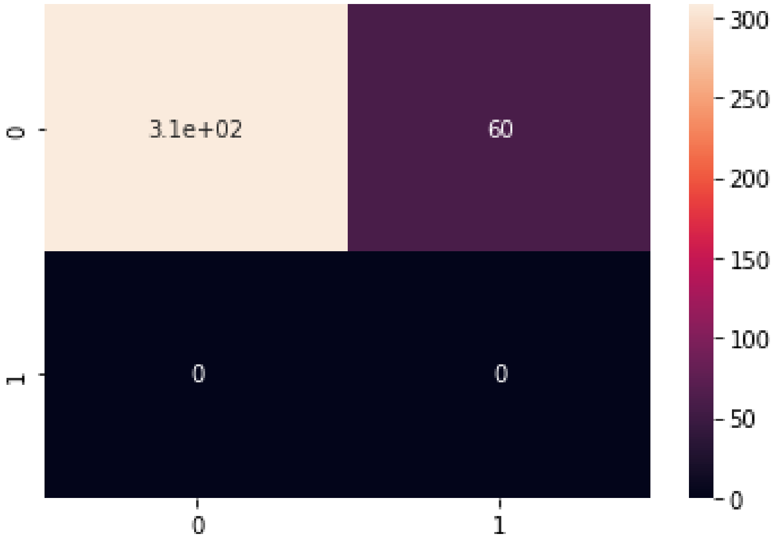


Table 8.5 Confusion Matrix for Random Forest Classifier

8.7 Gradient Boosting Classifier:

Gradient Boosting is an ensemble learning method used in machine learning to solve both regression and classification problems. Here’s how gradient boosting works: The first weak model is trained on the training data and makes predictions. The prediction errors of the first model are calculated and used to train the next weak model in the sequence. This process continues for several iterations, with each subsequent model being trained to correct the errors made by the previous models. The final model is a combination of all of the weak models, and its predictions are made by summing the predictions of all of the weak models.

Accuracy 86.68475217391305 %

	Precision	Recall	F1-Score	Support
0	0.84	1	0.91	305
1	0	0	0	60
Accuracy			0.84	368
Macro Avg	0.42	0.5	0.46	368
Weighted Avg	0.70	0.84	0.76	368

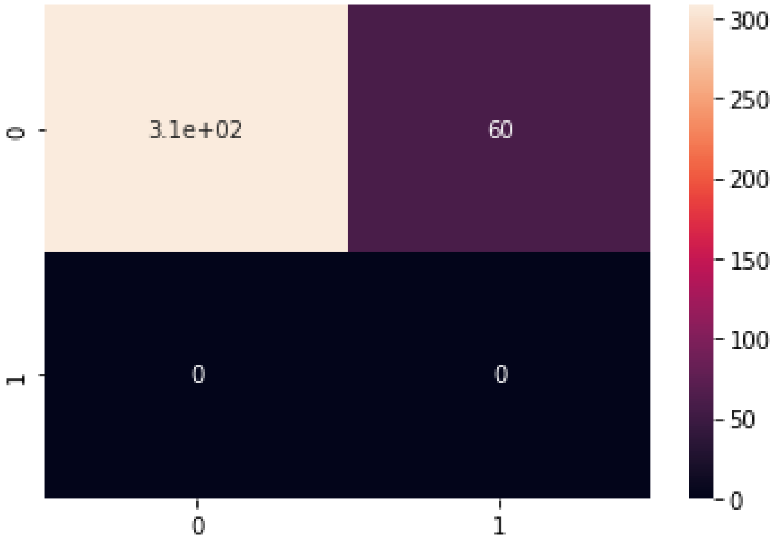


Table 8.5 Confusion Matrix for Gradient Boosting Classifier

8.8 XGBoost Classifier:

XGBoost (extreme Gradient Boosting) is a powerful and popular machine learning algorithm used for both classification and regression problems. XGBoost is an implementation of gradient boosting that is specifically designed to handle large-scale data and improve the performance of gradient boosting algorithms. Gradient Boosting is an ensemble learning method that combines the predictions of multiple weak models to produce a strong and more accurate model. In gradient boosting, each weak model is trained on the errors made by the previous model in the sequence, making it a sequential learning process. XGBoost uses decision trees as its base model, but it also incorporates several other techniques to improve the performance and scalability of gradient boosting algorithms.

Accuracy 86.68475217391305 %

	Precision	Recall	F1-Score	Support
0	0.84	1	0.91	305
1	0	0	0	60
Accuracy			0.84	368
Macro Avg	0.42	0.5	0.46	368
Weighted Avg	0.70	0.84	0.76	368

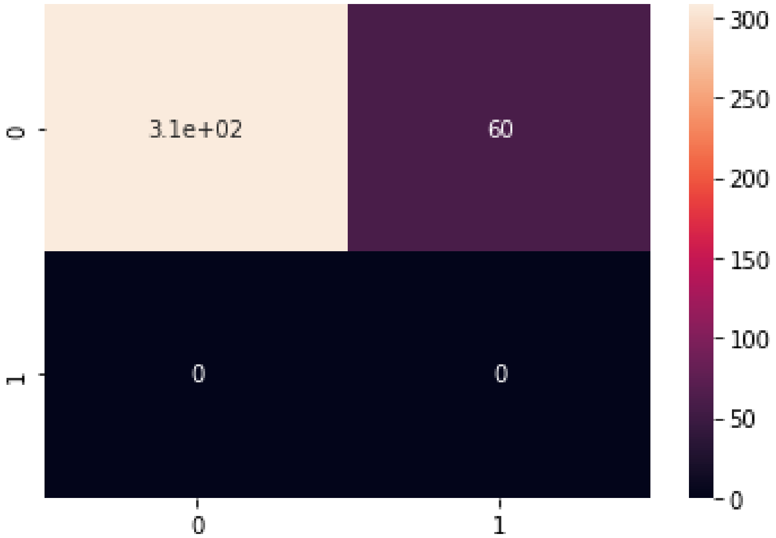


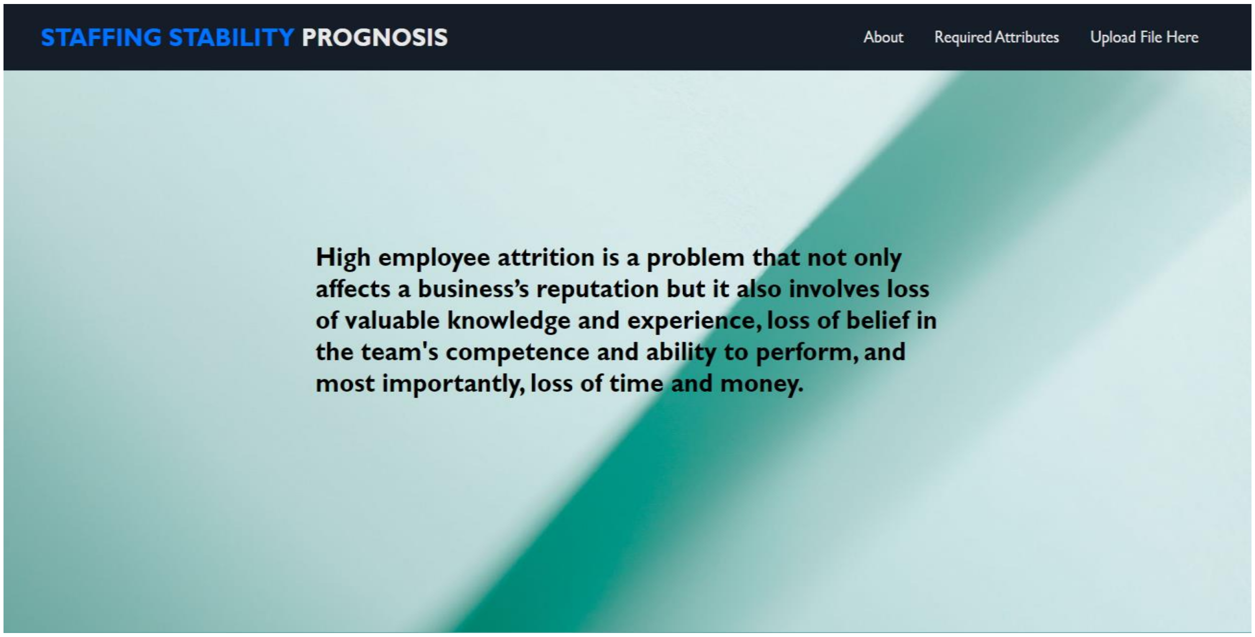
Table 8.5 Confusion Matrix for XGBoost Classifier

CHAPTER 9

SCREENSHOTS

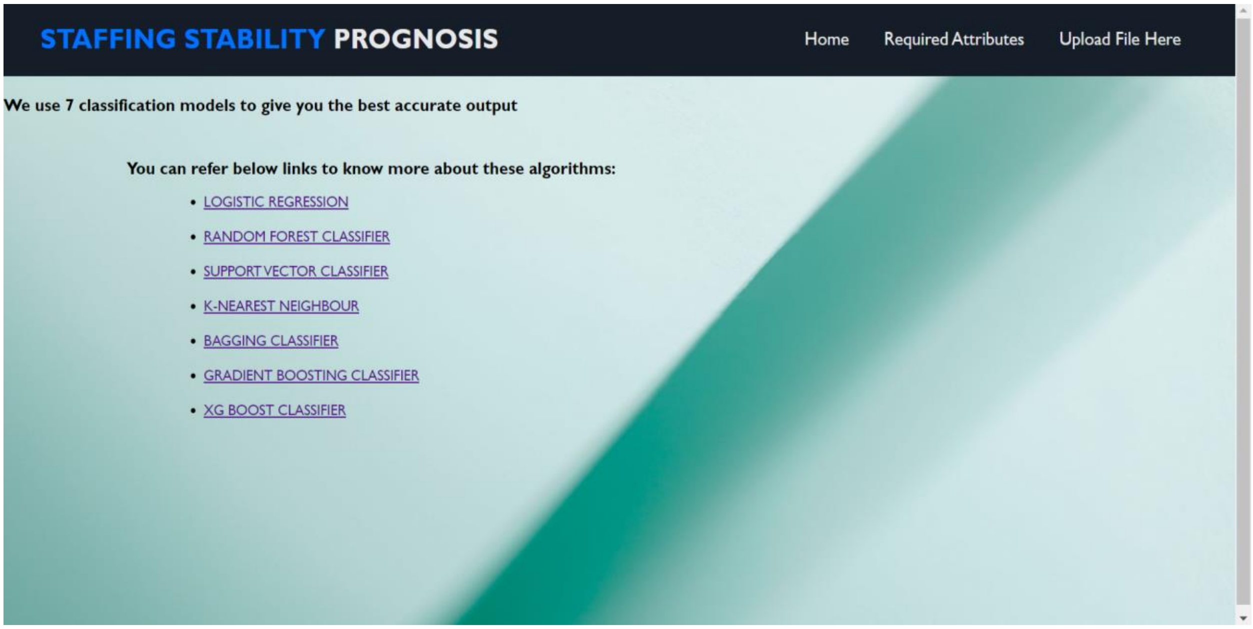
9.SCREENSHOTS

9.1 First User Screen:



High employee attrition is a problem that not only affects a business's reputation but it also involves loss of valuable knowledge and experience, loss of belief in the team's competence and ability to perform, and most importantly, loss of time and money.

9.2 Description about 7 Classification models

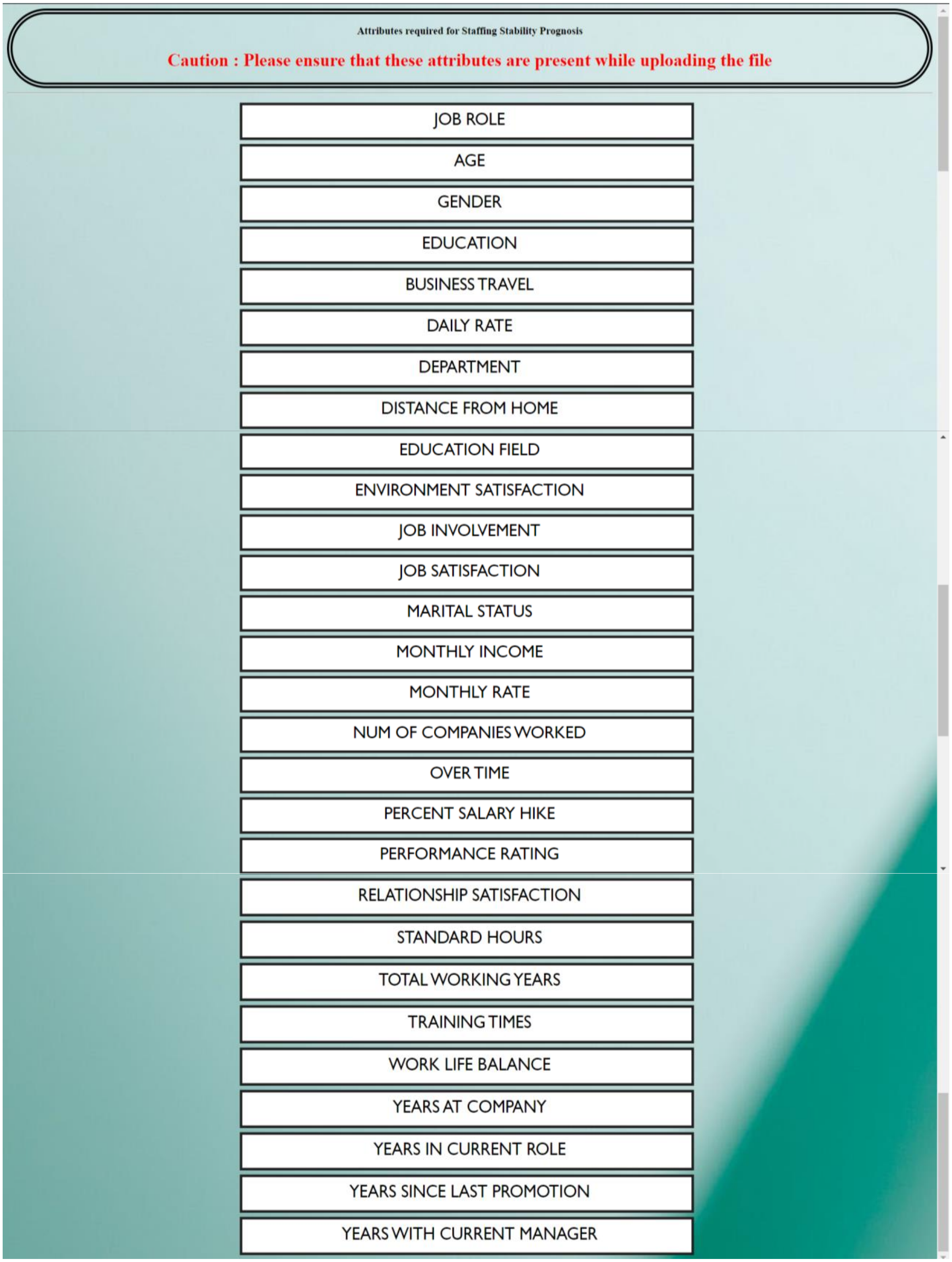


We use 7 classification models to give you the best accurate output

You can refer below links to know more about these algorithms:

- LOGISTIC REGRESSION
- RANDOM FOREST CLASSIFIER
- SUPPORT VECTOR CLASSIFIER
- K-NEAREST NEIGHBOUR
- BAGGING CLASSIFIER
- GRADIENT BOOSTING CLASSIFIER
- XG BOOST CLASSIFIER

9.3 Attributes required for Staffing Stability Prognosis



Attributes required for Staffing Stability Prognosis

Caution : Please ensure that these attributes are present while uploading the file

JOB ROLE

AGE

GENDER

EDUCATION

BUSINESS TRAVEL

DAILY RATE

DEPARTMENT

DISTANCE FROM HOME

EDUCATION FIELD

ENVIRONMENT SATISFACTION

JOB INVOLVEMENT

JOB SATISFACTION

MARITAL STATUS

MONTHLY INCOME

MONTHLY RATE

NUM OF COMPANIES WORKED

OVER TIME

PERCENT SALARY HIKE

PERFORMANCE RATING

RELATIONSHIP SATISFACTION

STANDARD HOURS

TOTAL WORKING YEARS

TRAINING TIMES

WORK LIFE BALANCE

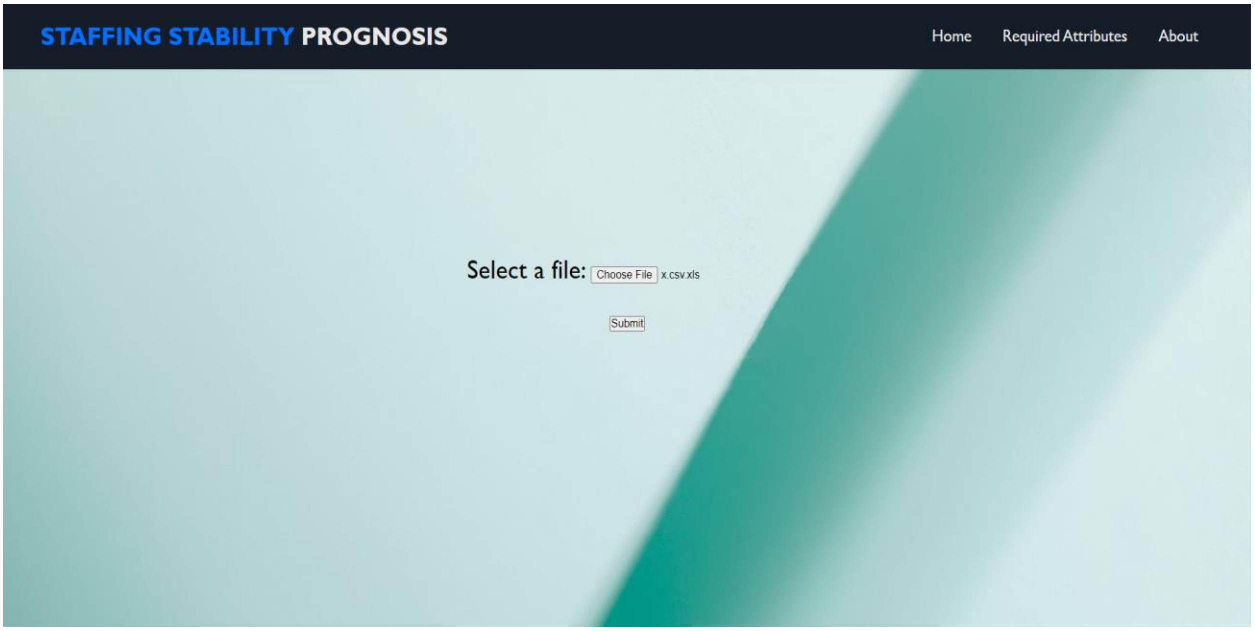
YEARS AT COMPANY

YEARS IN CURRENT ROLE

YEARS SINCE LAST PROMOTION

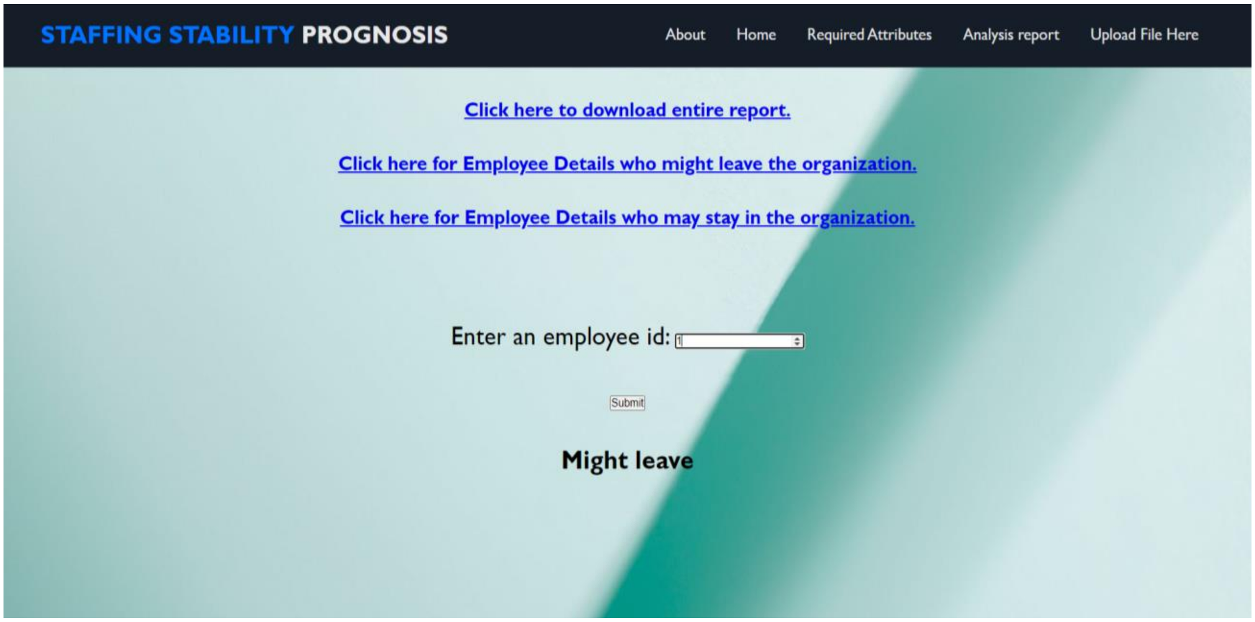
YEARS WITH CURRENT MANAGER

9.4 User Screen for uploading the dataset



Select a file: x.csv.xls

9.5 After Uploading the dataset, we can enter employee id to find whether that employee might leave or not.



Click here to download entire report.

Click here for Employee Details who might leave the organization.

Click here for Employee Details who may stay in the organization.

Enter an employee id: 1

Might leave

CHAPTER 10

CONCLUSION

10.CONCLUSION

Retaining an employee is one of the primary concerns to ensure company growth. The goal of the study is to compare the seven different machine learning techniques by using performance metrics. The model was evaluated using performance metrics such as accuracy, precision, recall, and F1-score, and it was found to have high accuracy and precision in Gradient Boosting Classifier with 85.93%. The study indicates that the use of such models can help organizations in predicting staffing stability and making informed decisions in their workforce planning and management.

CHAPTER 11

BIBLIOGRAPHY

11.BIBLIOGRAPHY

[1] Alao, D., Adeyemo, A.B.: Analyzing employee attrition using decision tree algorithms.Comput. Inf. Syst. Dev. Inform. Allied Res. J. 4 (2013)

[2] https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

[3] P. C. Bryant and D. G. Allen, “Compensation, Benefits and Employee Turnover: HR Strategies For Retaining Top Talent,” Compensation and benefits review, vol. 45, issue 3, pp. 171-175, May 2013.

[4] M. Panjasuchat and Y. Limpiyakorn, “Applying Reinforcement Learning for Customer Churn Prediction,” Journal of Physics. Conference Series, vol. 1619, issue 1, pp.

[5] J. Brownless, “4 Types of Classification Tasks in Machine Learning,” Machine Learning Mastery, Aug. 2020. [Online]. Available: https://machinelearningmastery.com/types-of classification-in-machine-learning/.

[6] Ronaldo C. Prati., Gustavo E. A., Batista.P. , Maria Carolina Monard , “A Survey on Graphical Methods for Classification Predictive Performance Evaluation”, IEEE Transactions on Knowledge and Data Engineering. 23, 2011