KADIR HAS UNIVERSITY
SCHOOL OF GRADUATE STUDIES
DEPARTMENT OF ADMINISTRATIVE SCIENCES

FORECASTING EMPLOYEES' PROMOTION BASED ON THE PERSONAL INDICATORS BY USING A MACHINE LEARNING ALGORITHM

YASMINE AYA IBRIR

MASTER OF SCIENCE THESIS

ISTANBUL, JUNE 2022

FORECASTING EMPLOYEES' PROMOTION BASED ON THE PERSONAL INDICATORS BY USING A MACHINE LEARNING ALGORITHM

YASMINE AYA IBRIR

A thesis submitted to
the School of Graduate Studies of Kadir Has University
in partial fulfillment of the requirements for the degree of
Master of Science in
Management Information Systems

Istanbul, June 2022

APPROVAL

FORECASTING EMPLOYEES' PROMOTION BASED ON THE PERSONAL INDICATORS BY USING A MACHINE LEARNING ALGORITHM submitted by YASMINE AYA IBRIR, in partial fulfillment of the requirements for the degree of Master of Science in

Management Information Systems is approved by

Dr. Mahmut Çavur (Advisor) ……..
Kadir Has University

Dr. Emrullah Fatih Yetkin ……..
Kadir Has University

Dr. Oğuzhan Ceylan ……..
Marmara University

I confirm that the signatures above belong to the aforementioned faculty members.

_____

Prof. Dr. Mehmet Timur Aydemir
Director of the School of Graduate Studies
Date of Approval: 23.06.2022

DECLARATION ON RESEARCH ETHICS AND PUBLISHING METHODS

I, YASMINE AYA IBRIR; hereby declare

- that this Master of Science Thesis that I have submitted is entirely my own work and I have cited and referenced all material and results that are not my own in accordance with the rules;
- that this Master of Science Thesis does not contain any material from any research submitted or accepted to obtain a degree or diploma at another educational institution;
- and that I commit and undertake to follow the "Kadir Has University Academic Codes of Conduct" prepared in accordance with the "Higher Education Council Codes of Conduct".

In addition, I acknowledge that any claim of irregularity that may arise in relation to this work will result in a disciplinary action in accordance with the university legislation.

Yasmine Aya Ibrir

_____

Date (23/06/2022)

To My Dearest Family...

ACKNOWLEDGEMENT

I am over helmed in all humbleness and gratefulness to acknowledge my depth to all those who have helped me to put these ideas, well above the level of simplicity and into something concrete.

I would like to express my sincere appreciation to my supervisor, Dr. Prof. Mahmut Çavur, for his guidance, support, encouragement, and positive attitude during my master's studies. As well as our university, which gave me the golden opportunity to do this wonderful thesis on the topic of FORECASTING EMPLOYEES' PROMOTION BASED ON THE PERSONAL INDICATORS BY USING A MACHINE LEARNING ALGORITHM, which also helped me in doing a lot of research and I came to know about so many new things. I am thankful to them.

I would also like to thank my committee members, Prof. Emrullah Fatih Yetkin and Prof. Oğuzhan Ceylan for serving as my committee members. I also want to thank you for letting my defense be an enjoyable moment, and for your brilliant comments and suggestions, thanks to you.

I am very grateful and would like to thank my family, for their invaluable patience, encouragement, endless support, and unconditional love.

I am making this project not only for marks but to also increase my knowledge.

Thanks again to all who helped me.

FORECASTING EMPLOYEES' PROMOTION BASED ON THE PERSONAL INDICATORS BY USING A MACHINE LEARNING ALGORITHM

ABSTRACT

Job promotion is considered one of the most important issues of importance in any organization, as it is vital for administrative development, and a means of motivating the worker for self-development and willingness to bear the burden and responsibility of work and the position attached to it, and thus it contributes to providing the necessary needs of the forces of mankind to occupy positions higher on the career ladder. Thus, this study aims to set up a sufficient framework to predict the promotion of an employee in an organization based on a variety of characteristics such as, but not limited to, the number of training, previous year rating, duration of service, awards earned, and average training score. Hence, this framework can be used and generalized to all prediction problems, not just our problem of predicting employee promotion. In this study, we used promotion data provided by Analytics Vidhya Data to test and prove the success of the framework. Our methodology is mainly composed of five phases: Input data, Data Pre-processing, Data Manipulation, Data Modeling, and finally Data Evaluation. We constructed a new number of features in this study. Then we used several features including creating features and providing insights into the promotion and commitment of employees and using supervised learning techniques, namely XGBoost, Random Forest, Decision Tree, Logistic Regression, AdaBoost, and Gradient Boosting. Experimental results show that the XGBoost model has a higher accuracy of 94%, proving to be the most efficient. The result is accentuated by the high validation score similar to accuracy and efficiency. It is a very important and valuable study as it is the first study to predict employee promotion using the XGBoost classifier method.

Keywords: Employee Promotion, Employee Promotion Prediction Framework, XGBoost, Machine Learning, Supervised Learning.

MAKİNE ÖĞRENİMİ ALGORİTMASI KULLANARAK KİŞİSEL GÖSTERGELERE DAYALI ÇALIŞAN TEŞVİKLERİNİN TAHMİNİ

ÖZET

İş terfii, idari gelişim için hayati önem taşıdığı ve çalışanın kendini geliştirmesi için motive etmenin bir aracı, işin yükünü ve sorumluluğunu ve bağlı olduğu pozisyonu üstlenmeye istekli olduğu için, herhangi bir organizasyonda en önemli konulardan biri olarak kabul edilir. Böylece kariyer basamaklarında daha üst sıralarda yer almak için insan kaynaklarının gerekli ihtiyaçlarının karşılanmasına katkıda bulunur. Bu nedenle, bu çalışma, eğitim sayısı, geçmiş yıl derecelendirmesi, hizmet süresi, kazanılan ödüller ve ortalama eğitim puanı gibi çeşitli özelliklere dayalı olarak bir çalışanın bir kuruluşta terfiini tahmin etmek için yeterli bir çerçeve oluşturmayı amaçlamaktadır. Bu nedenle, bu çerçeve sadece bizim çalışan terfiini tahmin etme problemimiz için değil, tüm tahmin problemlerinde kullanılabilir ve genelleştirilebilir. Bu çalışmada, çerçevenin başarısını test etmek ve kanıtlamak için Analytics Vidhya Data tarafından sağlanan promosyon verilerini kullandık. Metodolojimiz temel olarak beş aşamadan oluşur: Girdi verileri, Veri Ön İşleme, Veri Manipülasyonu, Veri Modelleme ve son olarak Veri Değerlendirme. Bu çalışmada yeni bir dizi özellik oluşturduk. Ardından, XGBoost, Random Forest, Decision Tree, Logistic Regression, AdaBoost ve Gradient Boosting gibi denetimli öğrenme tekniklerini kullanarak oluşturulan özellikler dahil olmak üzere çeşitli özellikler kullandık ve çalışanların terfi ve bağlılığına ilişkin içgörüler sağladık. Deneysel sonuçlar, XGBoost modelinin %94'lük yüksek bir doğruluğa sahip olduğunu ve en verimli olduğunu kanıtladığını gösteriyor. Sonuç, doğruluk ve verimliliğe benzer şekilde elde edilen yüksek doğrulama puanı ile vurgulanır. XGBoost sınıflandırıcı yöntemini kullanarak çalışan terfiini tahmin etmeye yönelik ilk çalışma olması sebebiyle de çok önemli ve değerli bir çalışmadır.

Anahtar Sözcükler : Çalışan Terfisi, Çalışan Terfi Tahmin Çerçevesi, XGBoost, Makine Öğrenimi, Denetimli Öğrenme.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONMYMS AND ABBREVIATIONS

AB	AdaBoost
AOC	Area under the ROC Curve
CV	Cross-Validation
DT	Decision Tree
FN	False Negative
FP	False Positive
FPR	False Positive Rate
GBM	Glioblastoma
HR	Human Resources
HRM	Human Resource Management
IT	Information Technology
KNN	K Nearest Neighbors
KPIs	key Performance Indicator
LR	Logistic Regression
min/max	minimum and maximum
ML	Machine Learning
RF	Random Forests
ROC	Receiver Operating Characteristic Curve
SMOTE	Synthetic Minority Oversampling Technique
std	Standard Deviation
SVM	Support Vector Machines
TN	True Negative
TP	True Positive
TPR	True Positive Rate

1. INTRODUCTION

1. 1 Motivation

Promotion has always been an important point of research in several areas, including the field of human development. Rationalizing this has become even more important given that the effect of any establishment is linked to people's capacity and willingness to interact, which has resulted in people becoming more ambitious and diversified in their ambitions, as well as more engaged in career planning.

Job promotion is considered one of the most important issues in any organization, as it is vital for administrative development and a means of motivating the worker for self-development. The willingness of the worker to bear the burden and responsibility of work, and the position attached to it, contributes to providing the necessary needs to occupy positions that are higher on the career ladder. Nowadays, many organizations experience the problem of job promotion and professional stability. This topic has received the attention of many thinkers and researchers, and we find many studies concerned with promotion and stability.

The excitement and passion displayed by a company's staff is one of the driving aspects of a very successful organization. Wealth is undoubtedly a motivating stimulus for workers, but acknowledgment of hard effort is just as crucial, if not more so, and arguably the most visible method of thanking an employee for their efforts is to promote them.

Many employees often complain about the process of promoting staff in their respective societies. As a result, promotion is one of the fundamental forces that play an essential and critical part in the behavior of individuals, driving forward their desire to perform. It can be said that the ability of institutions to achieve their goals depends largely on the extent to which the administration succeeds in providing sufficient satisfaction and setting up a program of promotion according to objective criteria that allow achieving organizational effectiveness. It is critical for businesses to be able to forecast what will

happen to their client base and staff so that they may take the appropriate actions prior to the "promotion" process.

The term "promotion" is defined as a means of an employee's career advancement and development and is linked to the employee's level of performance. Employees are promoted based on their practical efficiency and loyalty in performing their jobs, as well as their number of years of service and the qualifications they obtained while working. However, if employee promotions are not based on the best of fundamentals, administrative corruption will occur because the right person is not placed in the right place. Further, it will push employees into administrative conflict and cause them to lose a sense of belonging to the organization, which then leads to the collapse of the organization. Most prior research has focused on examining the causes and elements influencing employee promotion and have attempted to forecast employee promotion using statistical methods and data analysis techniques. The essential idea is to promote the right man to the right place, and thus be the path to success for the institution. Favoritism and relatives must be excluded from these accounts.

In the present study, our aim was to set up a robust framework that can be used and generalized to all prediction problems, not just the problem of predicting employee promotion. It depends on the evaluation of the company's employees in terms of commitment to attendance and departure, dedication to work, sincerity in work, and an attempt to make every effort to improve performance at work and learn everything available to advance the general interest of the corporation. We examine a variety of factors, including conventional employee characteristics as well as employment experiences. All these features are needed to decide who the right person is to receive a promotion in the organization. Therefore, our research is what comes after post-hire, creating a project that will predict the right person to get promoted in an organization according to some features. These features will be determined with respect to the literature in this study. It is formulated as a binary classification problem that classifies employees as either "will promote" (promoted) or "will not promote" (not promoted). We believe that our system can be successfully used in choosing the appropriate employee according to the organizational hierarchy without fraud or prior knowledge, so that there

will eventually be an employee promoted according to specific criteria (performance and practical efficiency).

Based on supervised machine learning, many models have been presented for various years. The primary goal of the learning models, given a currently working employee, is to reliably anticipate the individual's promotion within a particular time period by evaluating historical features, defining performance requirements, and reviewing performance technically and personally. Therefore, in our study, we will use a policy for a promotion that applies to all workers who are eligible for the promotion. Employees may be upgraded only after their three-month training period has ended, and they are not subject to a performance improvement plan. Employees may be upgraded within the same division or department or to another. We will reward workers based on their performance and workplace behavior, utilizing these aspects from the Human Resources (HR) Analytics Vidhya data as well as some new ones. Furthermore, our findings show that the newly introduced qualities are more relevant in predicting employee advancement than the other features.

1. 2 Contributions and Organization of Thesis

The following are our contributions to this thesis:

Using machine learning methods, we create a framework for predicting employee promotion. Hence, this framework can be used and generalized to all prediction problems, not just our problem of predicting employee promotion. In this study, we used promotion data and cases to prove the success of the framework.

This research is based on an empirical study according to a paradigm of competitive promotional strategies that would evaluate the pool of competitive candidates, comparing those selected against the candidates that were passed. As an important indicator for the promotion process, we chose existing features and added new ones; metric of sum, total score, work fraction, work start year, years remaining to retire, and performance; some of them were already in the data, others were added, and some were modified before

being used. All these features are needed to decide who the right person is to get promoted in the organization; we only used publicly available information about the employees. We compare the experimental results with different baseline models and use evaluation performance metrics (accuracy, precision, recall, receiver operating characteristic curve (ROC), Area under the ROC Curve (AOC)).

The rest of the thesis is organized as follows:
Chapter 1 is the introduction of the study and explains the problem statement shortly. Chapter 2 presents an overview of relevant research on the prediction of employee promotion. Chapter 3 provides extensive background information on the data structures, techniques, and methodologies employed in this thesis. All steps of our technique in this study are detailed in full in Chapter 4. In Chapter 5, the outcomes of the experiments are presented and discussed. Finally, Chapter 6 summarizes and closes the thesis with closing remarks and suggestions for further research.

2. RELATED WORK

This chapter divides related research into two categories based on two sorts of "predicting": employee promotion and employee turnover. Sections 2.1 and 2.2 outline the research on predicting employee advancement and turnover, respectively. Given the amount of research in linked domains, recent studies related to this issue are summarized in this section in detail.

2. 1 Predicting Employee Promotion

Employee promotion refers to an employee's upward progress within the organization to a new or higher job position, tasks, and responsibilities. Promotion is an important step in the life cycles of both employees and organizations. It is a critical issue to choose the correct candidate for promotion at the right moment. According to many studies, several strategies rely on machine learning to overcome real-world challenges, particularly in human resource management, and employee promotion is one of them.

Human resource is the first source and the most critical essence of each company. Managers spend a great deal of time recruiting capable employees. Furthermore, they regularly spend additional resources on training staff because they are an important aspect of every business and have a big impact on its growth. As competition overheats, competition between different companies has become more intense. Promotion is an issue that both businesses and employees are concerned about. On the one hand, promotion is a strategy used by businesses to pick exceptional people and boost their competitiveness. Employee promotion, systems, and organizational performance have a good relationship (Chen, Hsu, and Wu 2012). On the other hand, it is an opportunity for employees to recognize their worth and get prospects for advancement. Advancement prospects have a higher impact on employee performance, as do leadership, job promotion, and work environment. These components work together to improve employee performance (Febrina 2017). The promotion has fundamental motivating value since it increases an employee's authority, power, and position within a company. It is regarded as an excellent policy to replace gaps in higher-level positions through internal promotions since such

advancements give encouragement and motivation to employees while also removing sentiments of stagnation and discontent (Li et al. 2021).

According to certain research, internal promotion in organizations is influenced by a variety of factors, including age (Long et al. 2018; Machado and Portela 2021; Li et al. 2021), gender, education background (Jantan and Hamdan 2010; Long et al. 2018), and job experience (DE PATER et al. 2009; Long et al. 2018).

Categorization is one of the most important tasks in data mining, which is used to extract knowledge from massive amounts of data. This technique is frequently utilized in a variety of sectors, although it has received less attention in human resources management. Using an employee's performance data, an experiment was carried out to illustrate the practicality of recommended classification techniques. In his experiment which used the much more popular classification methods of neural networks, decision trees, and nearest neighbors, the C4.5/J4.8 classifier had the greatest accuracy 79.49% (Jantan and Hamdan 2010).

Febrina's purpose for his study was to investigate the impact of leadership, job advancement, and job environment on employee performance. In this research, higher compensation is always followed by a job promotion, whereas increased compensation is always followed by an increase in experience in problem-solving, loyalty, honesty, and responsibility at work. Based on his findings, it is possible to infer that leadership, job advancement, and job environment all have a positive and substantial influence on staff performance at the bank. These components work together to improve employee performance (Febrina 2017).

Liu et. al. emphasize that in the age of big data and Industry 4.0, businesses must prioritize human capital. Enterprises should utilize big data to study personnel, anticipate the future, and assist companies based on data collection. Their study provides insights for promotion research and intelligent human resources management research in the age of big data and Industry 4.0. Statistics, networks, and machine learning were utilized to investigate the relationship between organizational rank and advancement. It contends

that people should work in various places and divisions to broaden their experiences. Supervised learning is used to predict staff advancement. To build models, we use logistic regression (LR), random forests (RF), and AdaBoost (AB). In the end, RF performs the others and has a reasonable time consumption. In summary, working in jobs where mobility is more reliable, resources are available, or particular experience is accessible can help the worker advance (Liu et al. 2019).

Long et. al. construct characteristics and applies four machine learning approaches: logistic regression, random forests, Support Vector Machines, and AdaBoost. These are used to predict employee advancement using data from a Chinese state-owned firm. By extracting personal basic information and position information from this data, two types of features were created based on five methodologies, and their usefulness in anticipating employee promotion was subsequently tested. The RF model is used to determine the Gini significance of each feature. The bigger the value, the greater the influence on the forecast. Furthermore, it displays the ranking of 15 original characteristics based on their relative significance to anticipated values for a particular collection of categorical features. Correlation analysis is used to establish the impact of working years, gender, and the number of different roles in the promotion. According to the findings of the study, the influence of post features on promotion is stronger than that of personal basic characteristics. Then, using correlation analysis, they validate the efficiency of attributes in estimating employee advancement. Finally, the random forest model's prediction effect is determined to be comparatively better through model learning and testing (Long et al. 2018).

Sarker et. al. provide an overview of k-means clustering and decision tree algorithms. Job classifications are frequently established using the k-means clustering technique, which is a common method of job classification establishment. In this case, they used the k-means clustering technique to divide employees into separate clusters based on their performance quality. They utilized a decision tree algorithm to swiftly identify employees and make suitable decisions. Several steps were being taken in this situation to avoid any risk associated with employing a poor performer. Support Vector Machines (SVM), Random Forest, Naive Bayes, Neural Networks, and Logistic Regression were employed

in this research study to construct a model that delivers insights regarding employee performance and commitment. Employees are divided into three groups based on their level of performance. According to the results, support vector machines outperform the other classifiers in terms of accuracy (Sarker et al. 2018).

Although substantial progress has been made using big data analytic technologies in human resource management, research on the mining of promotion characteristics is limited, and further research is needed. Thus, using data from Analytics Vidhya, we build various promotion attributes and predict using machine learning methods.

2. 2 Predicting Employee Turnover

Employee turnover is seen as a critical issue for all firms these days; to address this issue, organizations are now relying on machine learning approaches to forecast employee turnover. Employee turnover can be viewed as a defacement of the organization's intellectual capital. The literature study focuses on the strategies and techniques provided by various researchers for forecasting employee attrition.

Jain and Nayyar divide the main reasons for employee attrition into six branches: Frozen Promotions and Salary Hikes, Lack of Decision-Making Ability, Imbalance of Work-Life, Employee Misalignment, Unsuitable Behavior, and Inadequate Professional Skills. The researchers proposed a new model for forecasting employee attrition based on machine learning using XGBoost. It is recognized as a superior algorithm in terms of memory use efficiency, accuracy, and running time. It is a very robust and scalable strategy for dealing with all types of noise in large data sets. The model provided in this research has a very low rate of less than 30% and an accuracy of around 90%. A total of 14 factors have a greater effect on the attrition rate than any other component. The XGBoost-based model performed the best, with a high specificity rate and a low error rate. It outperformed the baseline model in terms of accuracy, increasing it to 89% (Jain and Nayyar 2018).

In the case of employee attrition, an estimate was made as to whether or not the person will quit the organization. Using this approach, the business may choose the individuals

who have the highest likelihood of leaving the organization and then provide them with specific incentives. In this work, various machine learning techniques have been implemented; DT, RF, and SVM. Based on the findings of this study, it is possible to infer that RF outperforms. This study also tries to provide some insight into the many elements driving worker attrition and their potential answers (P. K. Jain, Jain, and Pamula 2020).

In another study where the author's objective was to predict whether a certain employee will depart, a strategy for selecting features to reduce the dimension of the feature space was described. The recommended feature selection improves the predictor's performance. The 1-max-out method was used to identify which characteristics should be removed. This paper offers a three-stage method for constructing an accurate employee attrition prediction model, which includes pre-processing, processing, and post-processing. This paper offers a three-stage method for constructing an accurate employee attrition prediction model, which includes pre-processing, processing, and post-processing. The parameters of the logistic regression model are validated by assessing their fluctuations when trained through several bootstraps. The results suggest that the "max-out" feature selection strategy improves the F1-score performance metric (Najafi-Zangeneh et al. 2021).

A research study insists that employee turnover in the Information Technology (IT) industry is significant. Often, their early attrition is the result of company-related or personal concerns. Therefore, the Random Forest classifier was revealed to be the best model for predicting IT staff attrition. A correlation matrix in the form of a heatmap was developed to determine the essential factors that may affect the attrition rate. First, they found the critical aspects that influenced employees, resulting in future attrition. Second, using a few categorization models, they aimed to reliably predict those individuals who planned to depart the firm within the next two years. In the first scenario, they evaluated all target variable classes, but in the second case, they excluded employees who still expressed reservations about leaving a certain business soon. Furthermore, for feature selection, an R program called "caret" was utilized, which generates a report based on the value and relevance of the features in their dataset. Caret assisted in the rating of such

aspects. This technique aids in detecting the proper characteristics for their dataset when the features can be differentiated. The Random Forest classifier was the most efficient model in their research, with the best accuracy and recall value when compared to the other models (Bandyopadhyay and Jadhav 2021).

2. 3 Problem Statement

Although some achievements have been made by applying big data analysis technology in human resource management, research on the mining of promotion features is relatively sparse and there is a need for further study. Each of the papers cited above primarily used data mining techniques or machine learning models in Human Resource Management (HRM) to predict the turnover or promotions of employees. While cited studies substantiate the applications of machine learning in the HRM domain, especially using XGBoost in predicting the employee turnover field, none have applied XGBoost to predict the employees' promotion based on their characteristics, which could be mined from previous employee records. Whereas this is the first study on forecasting employee promotion using the XGBoost classifier approach, and we will improve XGBoost's accuracy from scratch. However, in harmony with these applications, this study strives to build a framework for predicting employee promotion from previous records that is in parallel to the criteria used for promotion evaluation using primarily the XGBoost model and other models. This framework can be used and generalized to all prediction problems, not just our problem of predicting employee promotion.

HR analytics is transforming the way human resources departments work, resulting in increased efficiency and improved overall performance. For years, human resources have used analytics. However, data collection, processing, and analysis have been primarily manual, and given the nature of human resources dynamics and HR key performance indicator (KPIs), the method has been restricted to HR, which is in charge of looking after employees' well-being and ensuring they are satisfied in their jobs. As a result, it is remarkable that HR departments have only realized the value of machine learning belatedly. With machine learning, we may use predictive analytics to identify employees who are most likely to be promoted based on prior data such as their degree, experience, age, ratings, and overall score (Analytics Vidhya dataset).

Employee promotions are a crucial component of keeping a motivated and skilled workforce. Employee promotion refers to an employee's advancement to higher ranks, and it is this aspect of the job that motivates workers the most. The ultimate reward for devotion and loyalty to a business is promotion, and the HR department plays a key role in managing all of these promotion responsibilities based on ratings and other accessible criteria. Thus, before the "promotion" process, firms must be able to predict what will happen to their client base and workforce. In our study, the essential goal is to promote the appropriate person in the proper position, which will lead to the institution's success. We consider several elements, including traditional employee traits as well as work history. Favoritism and family must be excluded from these accounts to prevent the problem from being passed on to someone else. We believe that, without fraud or prior information, our approach may be effectively utilized to select the proper individual according to the organizational hierarchy. Many models for various years are offered based on supervised machine learning. The learning models' main purpose is to accurately use and generalize our framework to all prediction problems, likewise, predicting an individual's promotion within the right position. In addition, the feature selection strategy aids in identifying the proper features in our dataset, allowing us to readily distinguish the characteristics that play a key part in promotion prediction issues; minimizing the number of input variables while creating a predictive model. In general, it is preferable to limit the number of input variables in order to reduce modeling computational costs and, in our situation, increase model performance – this is the uniqueness of our study. For these reasons, we will apply statistical methods to the independent variables and target variables one by one and rank them based on their results, with a higher score suggesting that the character is more significant or relevant to our output variable.

3. BACKGROUND

3. 1 Data Description

The dataset used in this study is provided by Analytics Vidhya Data Analysis. This dataset has 14 characteristics and 54808 records for train data and 23490 records for test data. Not all the 14 features are taken into account in our work for employees' predictions of promotions. We will choose the relevant aspects and add new ones that impact the employee's promotion as an important indication of the promotion process. Table 3.1 lists all the attributes, along with a thorough description of each.

Table 3.1 Attributes description.

Employee ID	Unique ID for the employee
Department	Department of the employee
Region	Region of employment (unordered)
Education	Educational level
Gender	Gender of the employee
Recruitment channel	Channel of recruitment for the employee
No. of trainings	No. of other trainings completed in the previous year on soft skills, technical skills, etc.
Age	Age of the employee
Previous year rating	Employee rating for the previous year
Length of service	Length of service in years
Awards won	If awards were won during the previous year, then 1 else 0
Avg training score	Average score in current training evaluations
KPIs met >80%	If the percentage of KPIs (Key Performance Indicators) >80%, then 1, else 0
Is promoted	(Target) Recommended for promotion

The dataset provides a target attribute, indicated by the variable *is promoted*. "No" denotes an employee who did not receive the promotion in a firm, and "Yes" represents an employee who did receive the promotion. This dataset will enable the machine learning system to learn from real-world data rather than explicit programming. The predictions made in the output will be more accurate if this training procedure is repeated over time and on relevant data. The dataset contains a target feature, identified by the variable *is promoted*. "No" represents an employee that did not get the promotion, and "Yes" represents an employee that got a promotion in a company. This dataset will allow the machine learning system to learn from real data rather than through explicit programming. If this training process is repeated over time and conducted on relevant samples, the predictions generated in the output will be more accurate.

3. 1. 1 Data Exploration

Data exploration is the most essential step in data analysis. The data is described using statistical and graphical approaches. We must first investigate the data in order to further evaluate it and highlight the relevant parts. Relation and correlation between each attribute may give some insight before processing data. During the data exploration phase, the methodologies of variable identification, univariate analysis, bivariate analysis, and multivariate analysis were applied to the Analytics Vidhya dataset step-by-step.

3. 1. 2 Variable identification

The initial stage in the data exploration process is variable identification. This procedure was carried out in two stages. In the first stage, predictor variables are identified as input variables, while target variables are identified as output variables. The next step is to identify the data type and category of the variables, as illustrated in Table 3.2.

Table 3.2 Variable identification attributes.

Attributes	Data Type	Variable Category	Type of Variable
Employee ID	Numeric	Continuous	Predictor
Department	Character	Categorical	Predictor
Region	Character	Categorical	Predictor
Education	Character	Categorical	Predictor
Gender	Character	Categorical	Predictor
Recruitment channel	Character	Categorical	Predictor
No. of trainings	Numeric	Categorical	Predictor
Age	Numeric	Continuous	Predictor
Previous year rating	Numeric	Categorical	Predictor
Length of service	Numeric	Continuous	Predictor
Awards won	Numeric	Categorical	Predictor
Avg training score	Numeric	Continuous	Predictor
KPIs met >80%	Numeric	Categorical	Predictor
Is promoted	Numeric	Categorical	Target variable

3. 2 Descriptive statistics

3. 2. 1 Uni-variate analysis

Univariate analysis is the most basic type of statistical analysis. Like other forms of statistics, it can be inferential or descriptive. The crucial point is that there is just one variable in action. When the multivariate analysis is more suitable, the univariate analysis might produce deceptive findings. Continuous and categorical variables are investigated in the univariate analysis. The variable type influences a strategy for doing univariate analysis (categorical or continuous). Individually, we investigated various techniques and statistical measures for categorical and continuous variables.

In this section, we will discuss continuous variables with a concentration on the variable's mean, standard deviation, and spread. Several statistical metrics and visualization approaches are used to describe this type of relation.

Table 3.3 Dataset descriptive statistics.

	employee_id	**no_of_trainings**	**age**	**previous_year_rating**	**length_of_service**	**KPI_met >80%**	**awards_won**	**avg_training_score**	**is_promoted**
count	54808.00	54808.00	54808.00	50684.00	54808.00	54808.00	54808.00	54808.00	54808.00
Mean	39195.83	1.253	34.803	3.329	5.865	0.352	0.023	63.386	0.085
Std	22586.58	0.609	7.660	1.259	4.265	0.477	0.150	13.371	0.279
Min	1	1	20	1	1	0	0	39	0
25%	19669.75	1	29	3	3	0	0	51	0
50%	39225.50	1	33	3	5	0	0	60	0
75%	58730.50	1	39	4	7	1	0	76	0
max	78298.00	10	60	5	37	1	1	99	1

At this step, we constructed the descriptive statistics of the dataset in order to examine the properties of all variables. We took into account the following variables: count, mean, standard deviation (std), minimum and maximum values (min/max), and 25%/50%/75% percentile. Figure 1 is an excerpt from the full dataset.

For a variable with a categorical value, the easiest approach to understanding the spread of each category variable is to use a frequency distribution. It is expressed as a percentage of the values in each category. Two measures may be used to assess it. Count the number of items in each category and the percentage of items in each category. A bar chart can be used as a tool for visualization.

Figure 3.1 represents the bar chart of the variables; these are the value ranges of all features. Every feature has a different distribution of values. We will look at each one independently to get a better and deeper understanding of the characteristics.

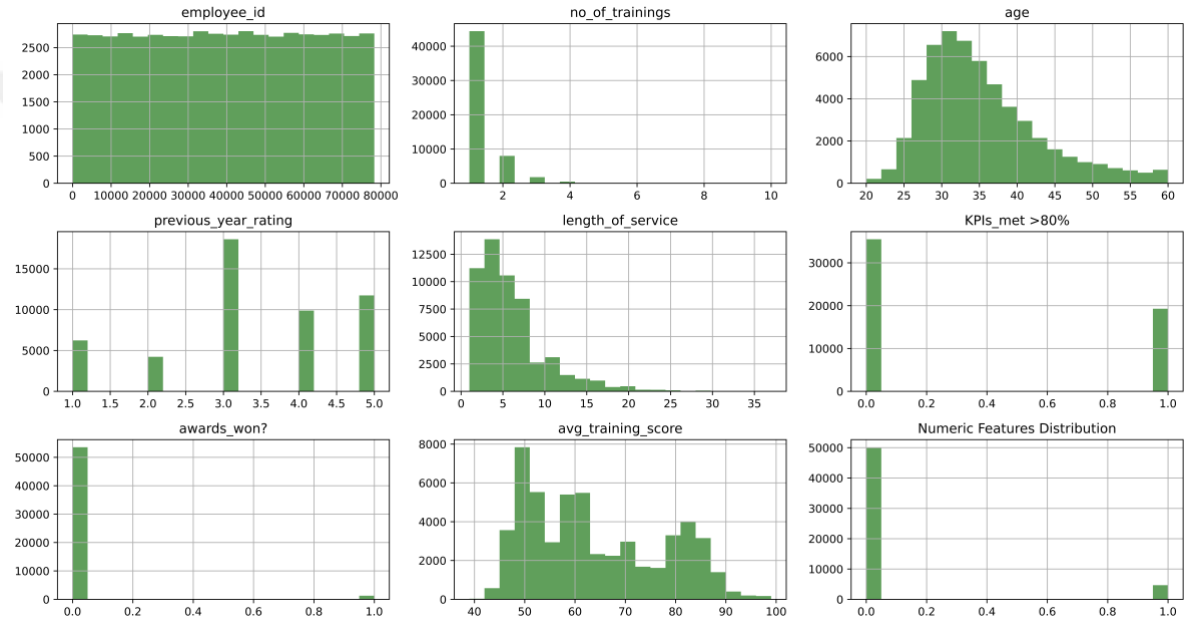
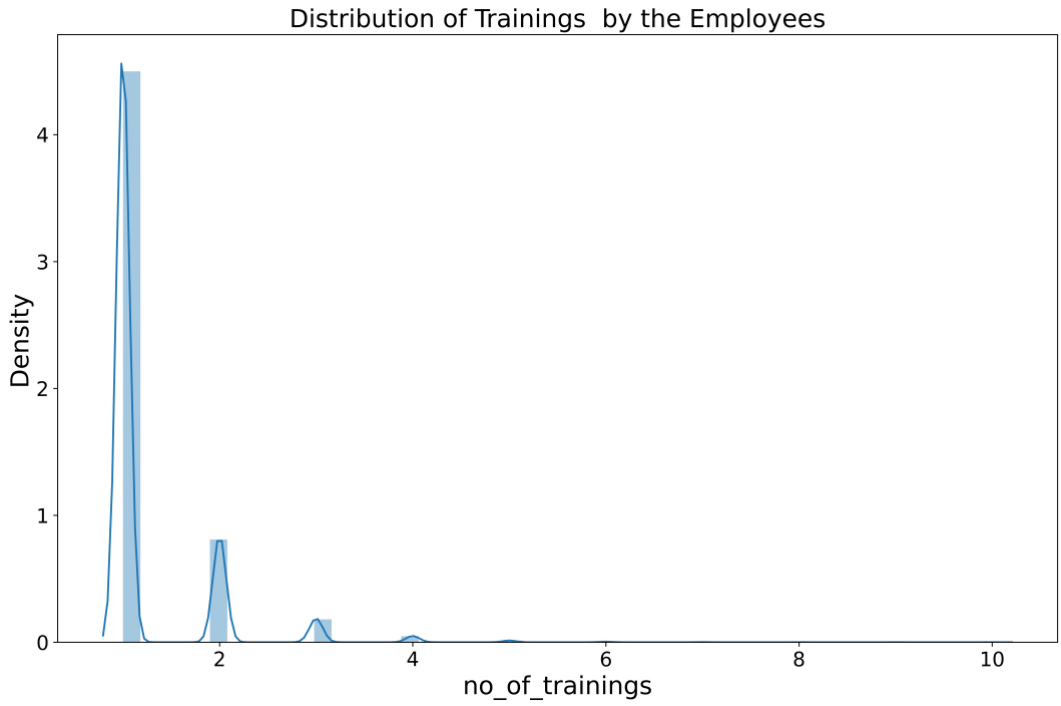


Figure 3.1 Numeric features distribution

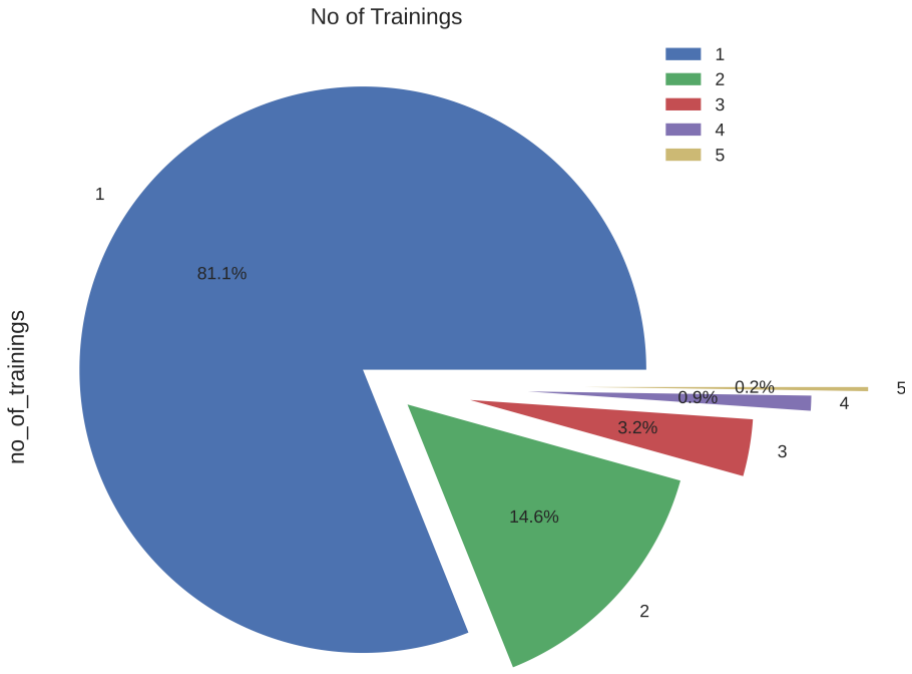
a. No. of Trainings Distribution

shows that the data is ordinal data that represents the number of trainings for the employee. As it is a number, it is given the integer datatype. There is no harm in changing it into an object. Therefore, the data is right-skewed and most of the employees have received 2 or 3 trainings, so we can see the peak at two. When we look at the number of trainings, we can see that 81% of employees attended one training. When we are checking the distribution of training undertaken by the employees, it is visible that 80% of the

employees have taken the training only once, and there is a negligible number of employees who have attended training more than three times.



a)

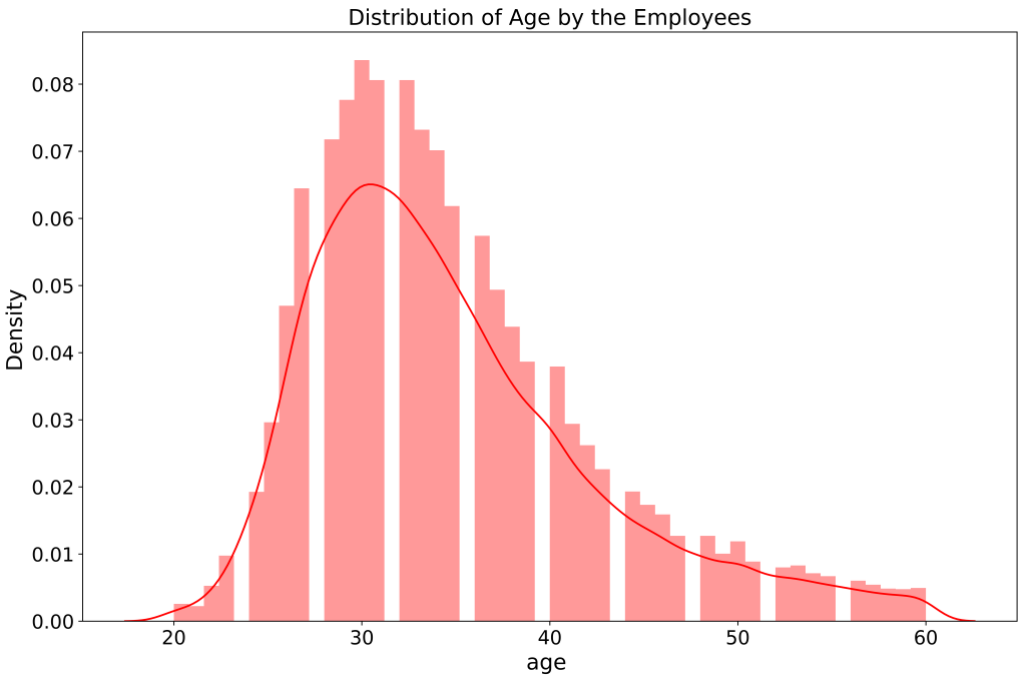


b)

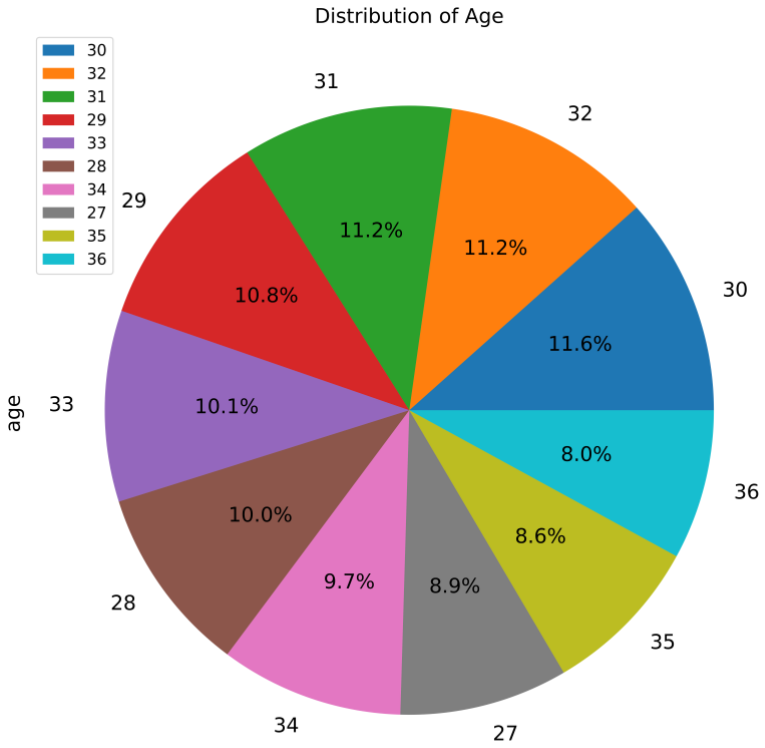
Figure 3.2 Distribution of Trainings by employees a) in number, b) in percentage.

b. Age Distribution

From our research, age will also play a crucial role in promotion, as the company has more employees in the age range of 25–38 who are young and hardworking. The data is near normal, but due to fewer employees in the higher age groups, it is likely to be right skewed as shown in Figure 3.3.



a)

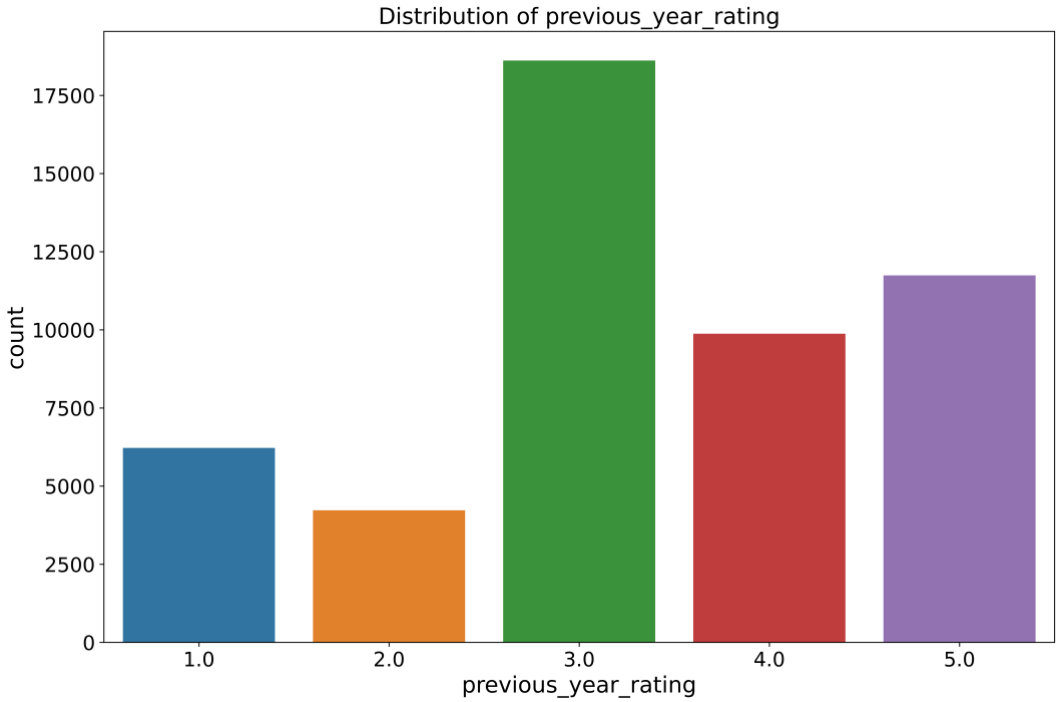


b)

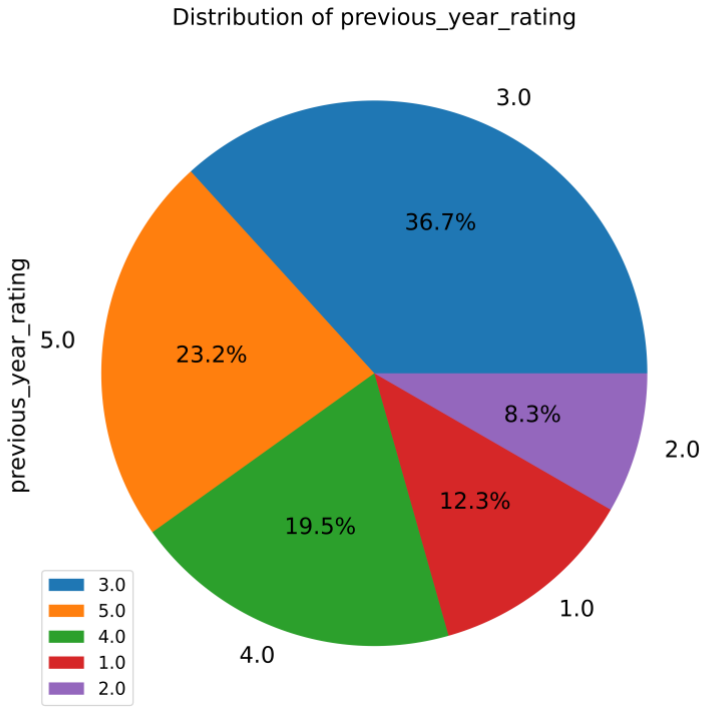
Figure 3.3 Distribution of Age by employees a) in number, b) in percentage.

c. Previous Year Rating Distribution

The previous year's rating describes the rating an employee received in the internal evaluations conducted by the company the previous year. These ratings give a clear differentiation between employees. Figure 3.4 displays that rating 3 is mostly used for the employees in all data with 36%, and the 5 rating with 23%. By intuition, an employee with a good rating paired with other factors is more likely to be promoted. To prove this hypothesis, the previous year's rating column is first encoded into labels for understanding. These rating labels are chosen to be subtle: New employee - rating 0, Minimum - rating 1, Fair - rating 2, Improving - rating 3, Good - rating 4, Very good - rating 5.



a)



b)

Figure 3.4 Distribution of Previous year's rating by employees a) in number, b) in percentage.

Figure 3.5 proves the hypothesis: employees with a grade of 'Very good' (5) are the most likely to get promoted. The trend in the number of people promoted is upwards, from a 'Minimum' to 'Very good' rating. "New" (no previous year rating) employees show a good percentage of people being promoted. For these employees, other factors are dominant.

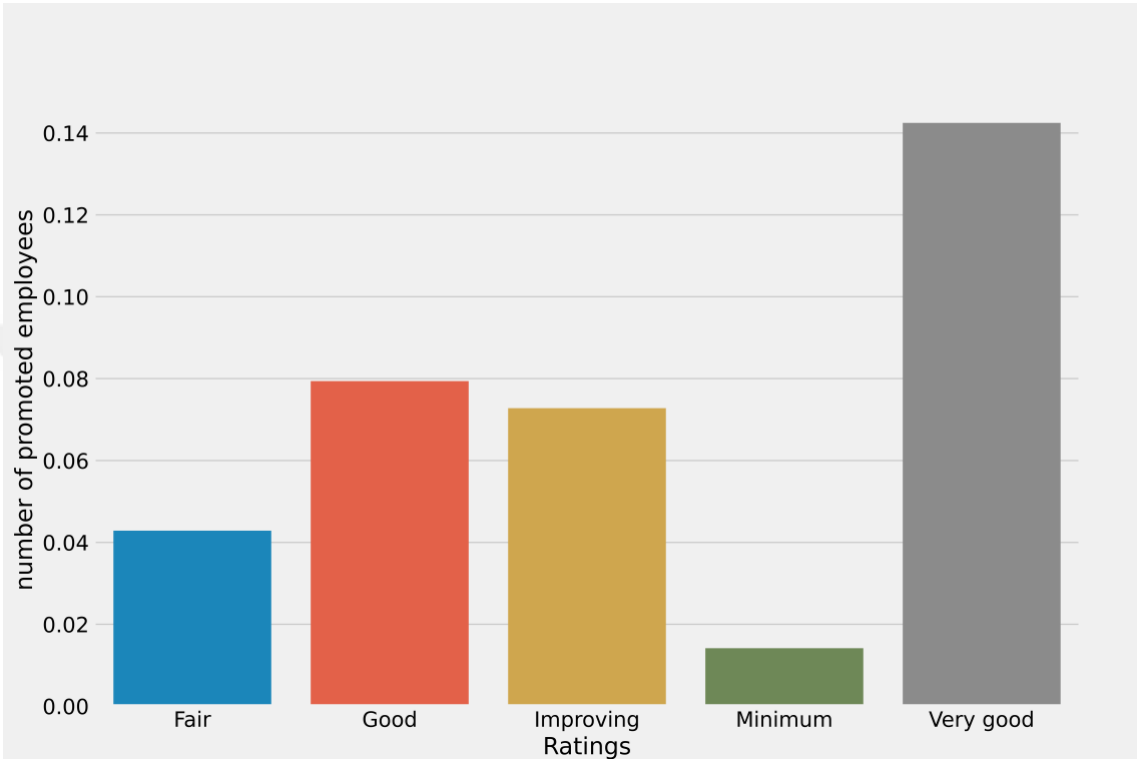
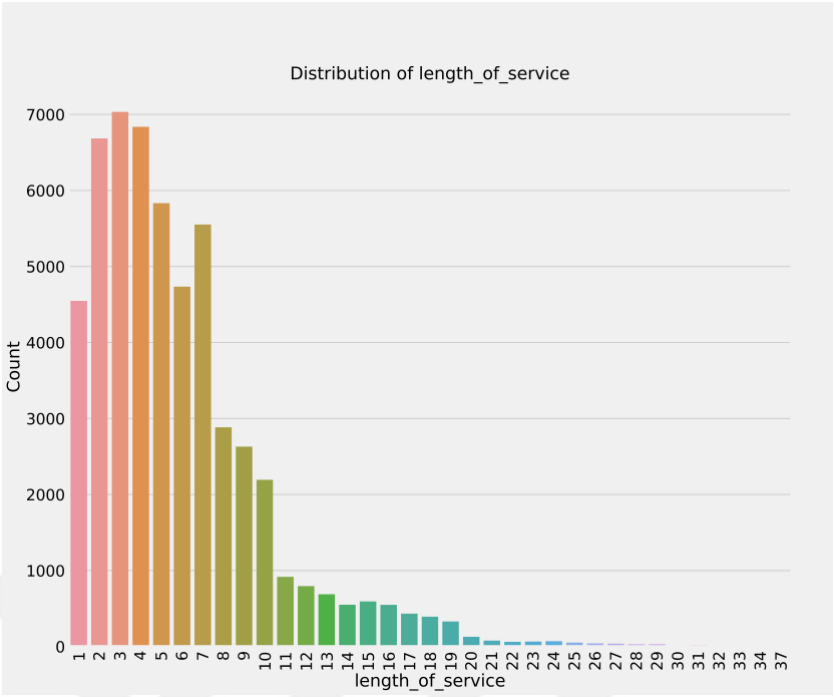


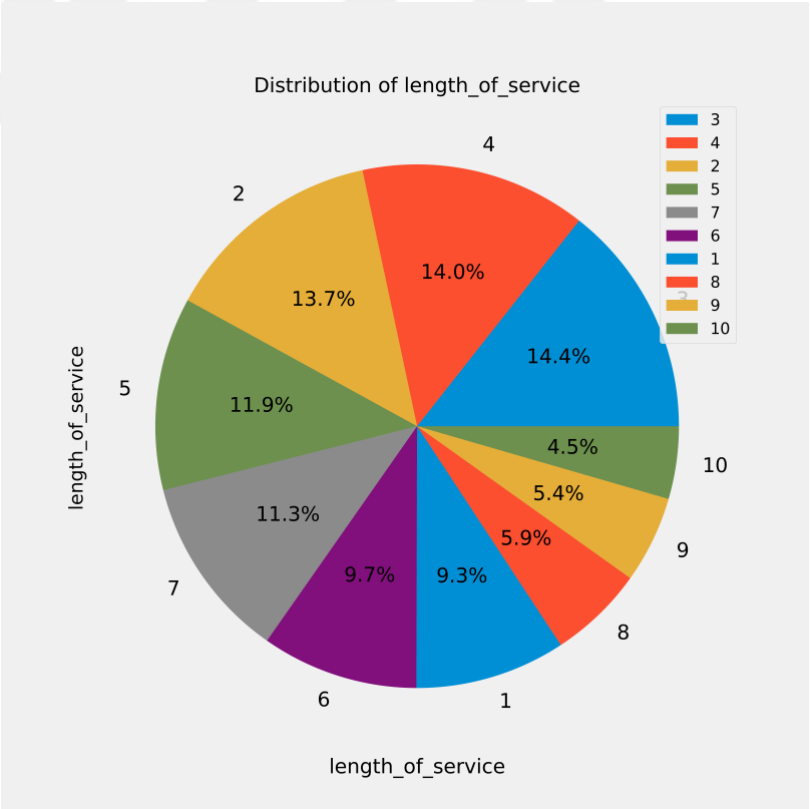
Figure 3.5 Distribution of ratings.

d. Length of Service Distribution

A similar approach is taken for the length of the service column in Figure 3.6. An employee who has been in a company longer is more likely to be promoted than a new hire. The length of service is from 1 to 10 years. After 10 years, there are few employees. Since the column is numeric, it is converted to categorical via binning into the following categories: New - 0 to 2 years, Established - 2 to 7 years, Experienced - 7 to 10 years, Veteran - 10 years or more. The categories have been chosen based on the general trend (logic).



a)



b)

Figure 3.6 Distribution of Length of service by employees a) in number, b) in percentage.

Figure 3.7 demonstrates that experienced employees are more likely to get promoted than other categories, due to their experience and understanding of the company. New and Established employees have an almost equal likelihood of getting promoted. Veteran employees generally get fewer promotions. Due to the number of years given to the company, the majority of them might have reached the pinnacle.

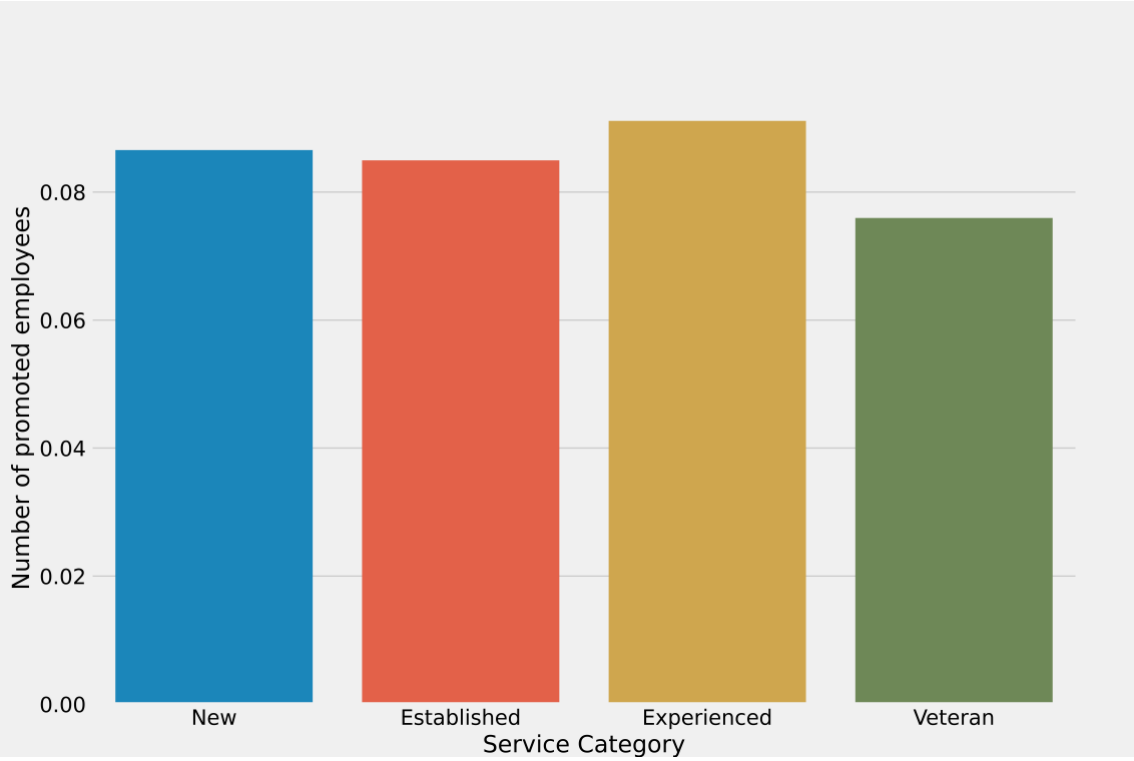


Figure 3.7 Service Category.

e. Awards Won and KPIs met >80 Distribution

In Figure 3.9 and Figure 3.9, only 2.3% of employees won awards, which is a very low ratio. Surprisingly, in the 'KPIs met >80' feature, more employees did not achieve this goal (0s) than employees who did (1s).

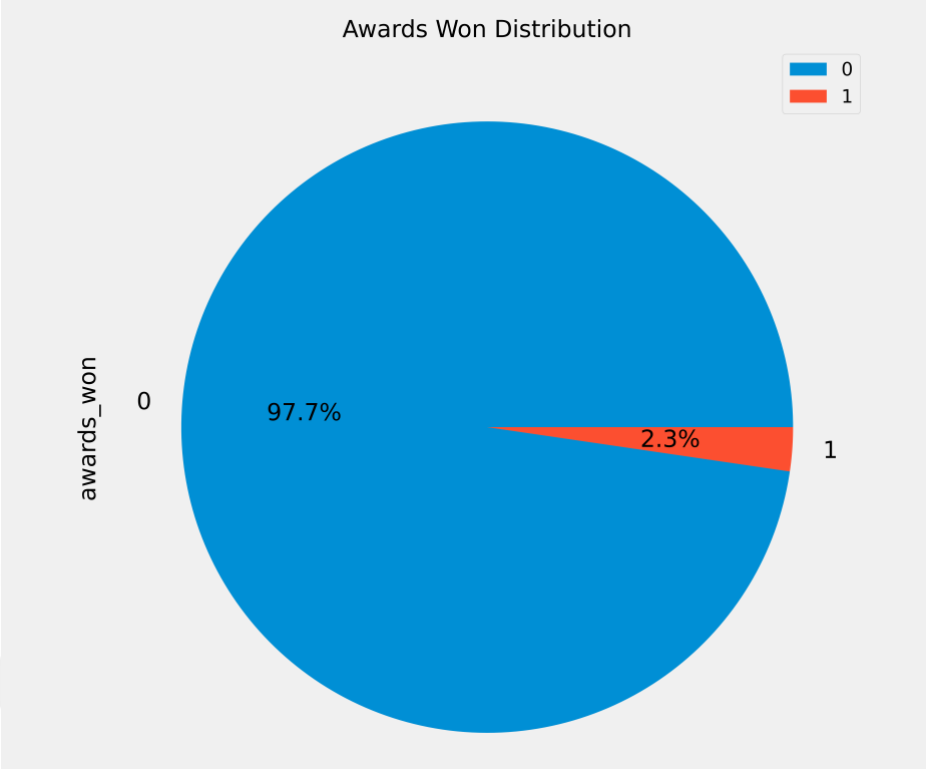


Figure 3.8 Distribution of Awards Won.

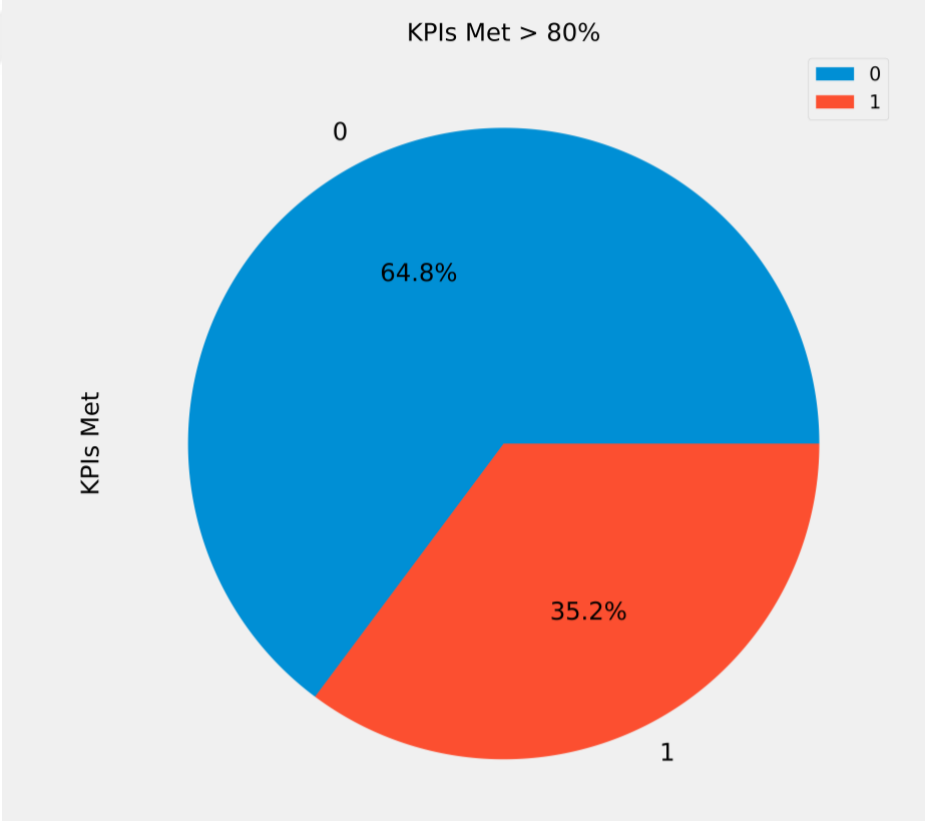


Figure 3.9 Distribution of and KPIs met by employees.

The KPIs met and awards won columns are inconclusive individually in the count plots. In Figure 3.10, the first two plots convey the distribution of awards won (left) and KPIs met (right) for the employees who were promoted. The third (bottom) plot shows the distribution of is promoted for all employees who have both won an award and met the KPIs. A lower percentage of employees who were promoted have won awards. An employee who meets the targets has a high chance of promotion. Employees who have both the award and the met targets are likely to be promoted.

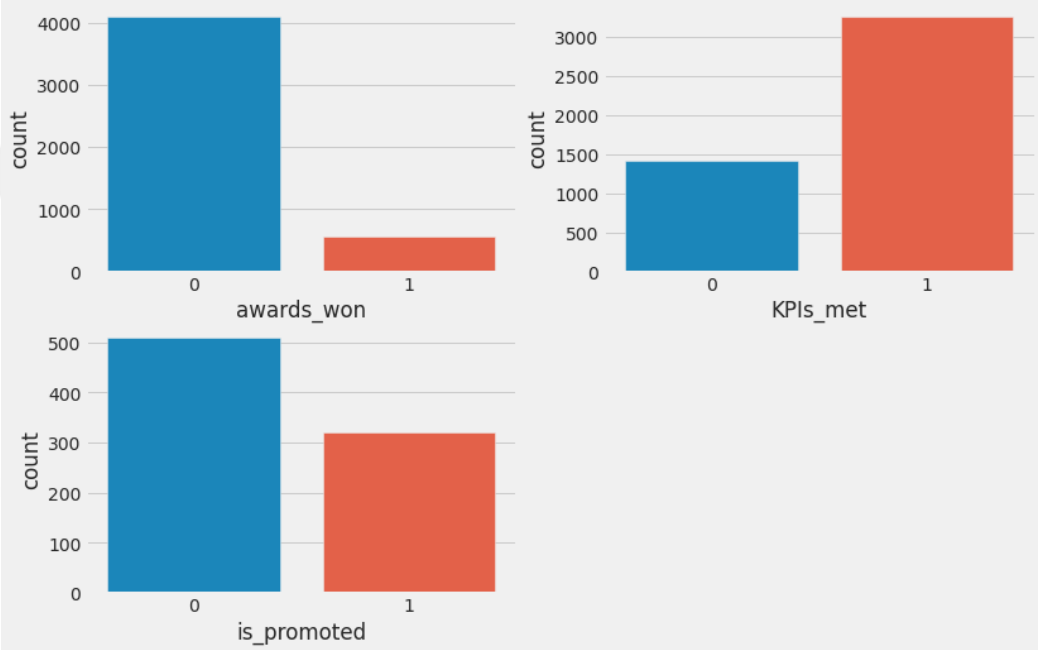
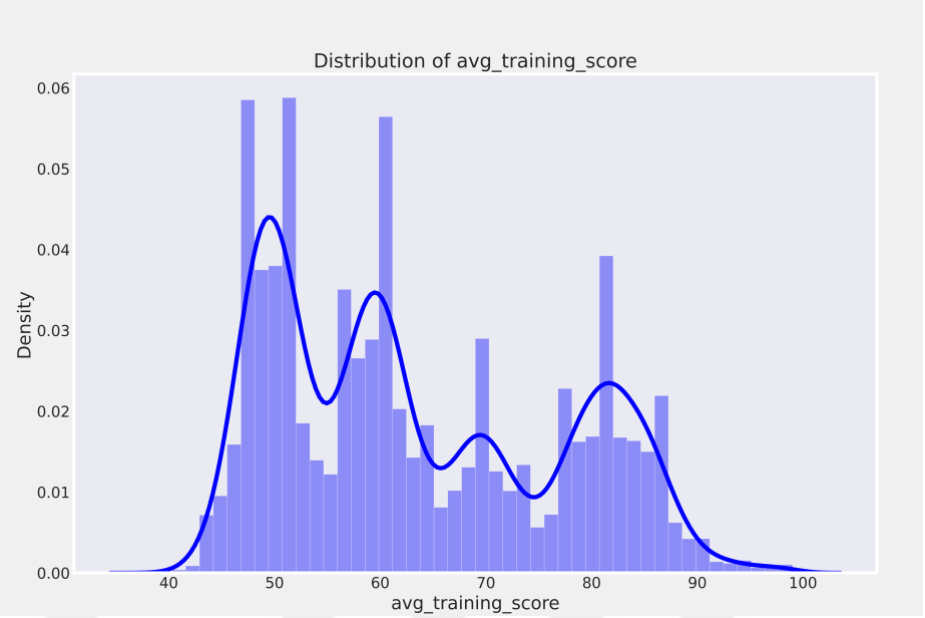


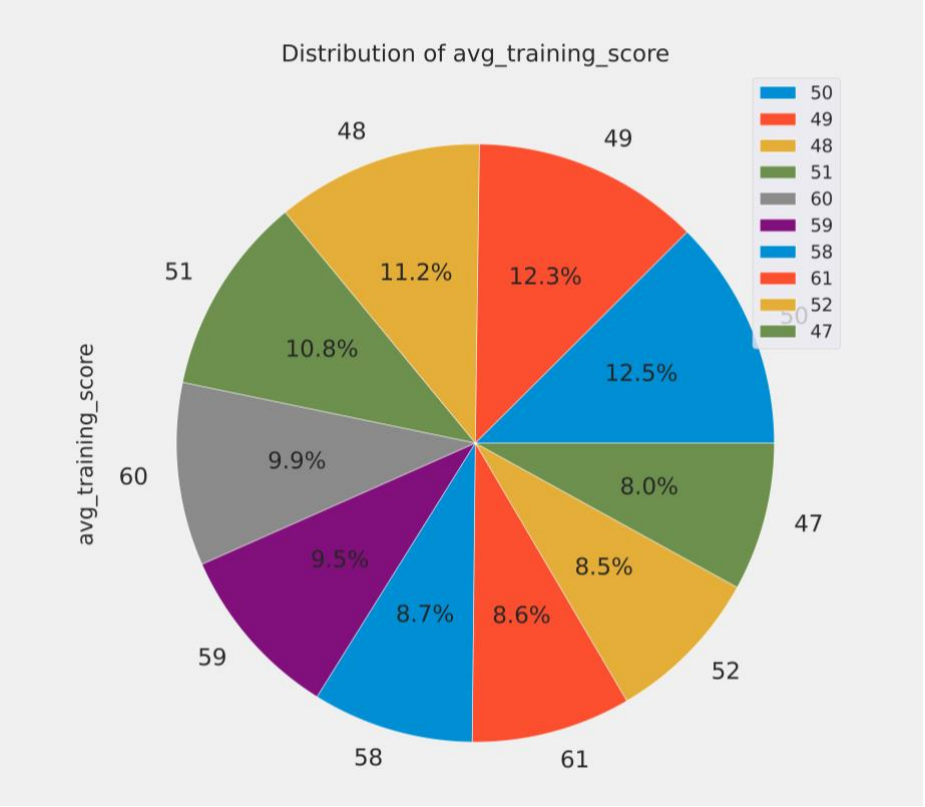
Figure 3.10 Distribution of Awards Won and KPIs met.

f. Average Training Score Distribution

The average training score is the average of all the training scores obtained, ranging from 40–to 100, and has different peaks as it is the average of all the scores. While most of the employees have scored in the range of 50-60, the lowest score bin has a very faint number as shown in Figure 3.11.



a)



b)

Figure 3.11 Distribution of Avg Training scores by employees a) in number, b) in percentage.

g. Department and Education Distribution

The next columns to analyze are the department and education in Figure 3.12 and Figure 3.13. Intuitively, the education of an employee is very important when recruiting, but once the employee has joined, their performance within the company is more important. Promotions happen within each department, meaning a sales employee is promoted but stays in the sales department. This can be considered generally true, with a few exceptions, of course. By these arguments, the two columns offer less insight when analyzing which factors affect an employee's promotion.

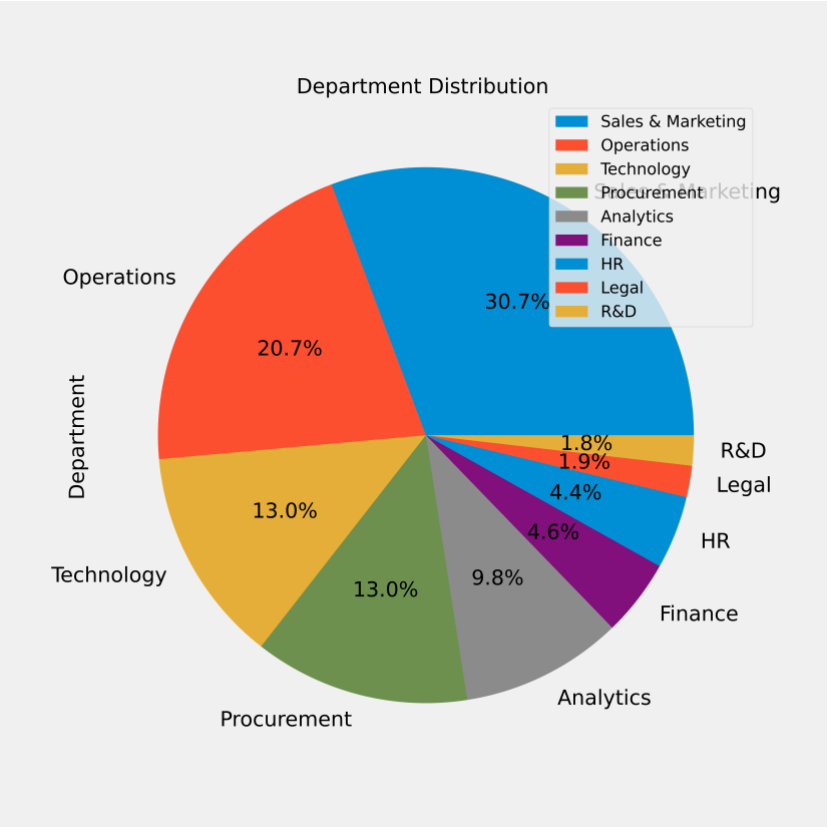


Figure 3.12 Department Distribution.

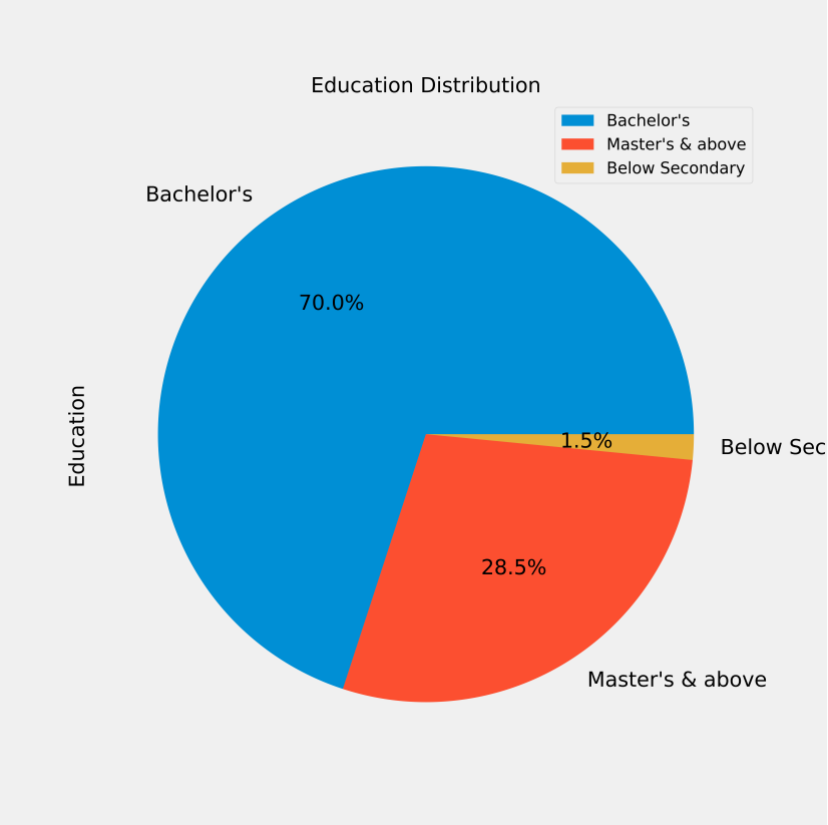
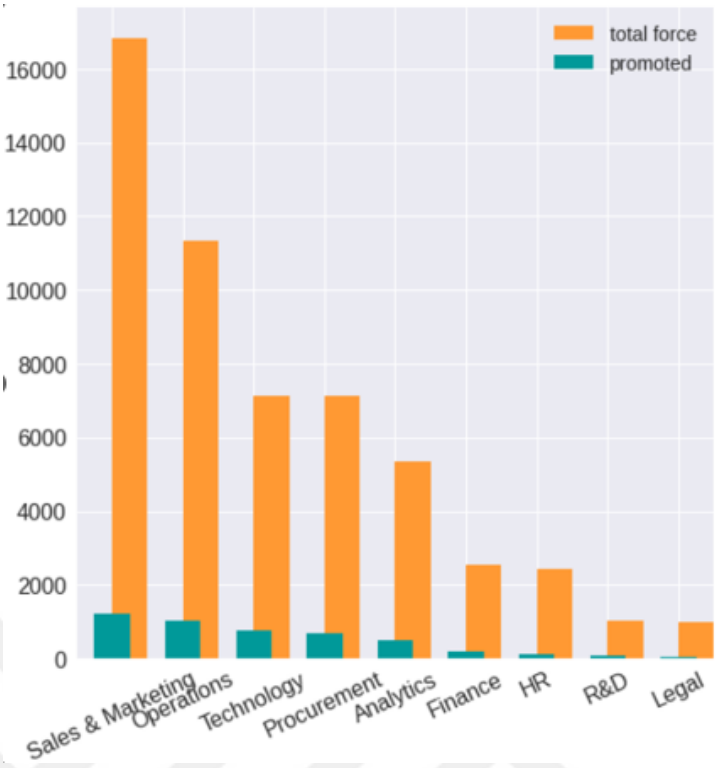
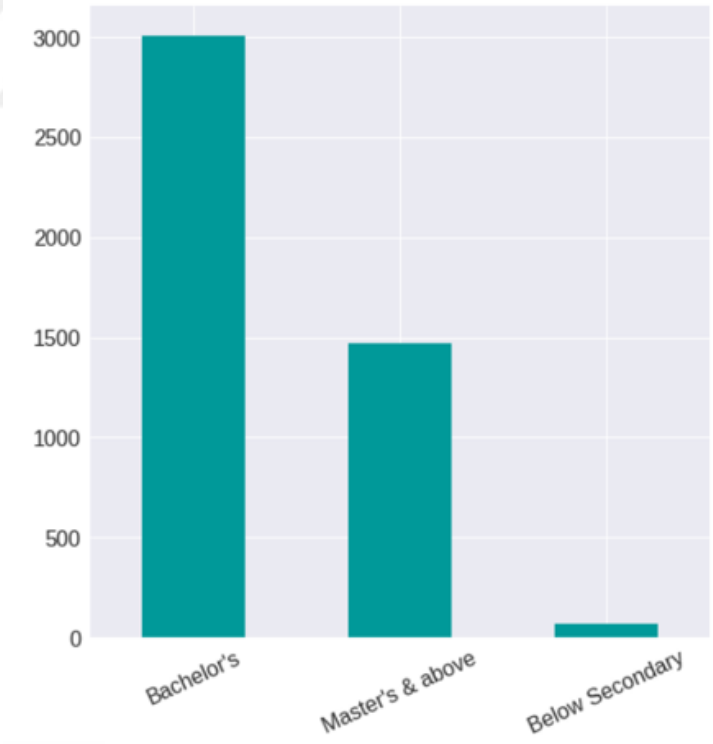


Figure 3.13 Education Distribution.

From Figure 3.14, it is obvious that the Sales and Marketing department has the highest percentage of promoted employees than other departments. It is 30% of the whole data, and the Operations department is 20%. Moreover, if we look at the education of employees, Bachelor's has 70% data, 28% Master's and 1.5 below secondary education.



a)



b)

Figure 3.14 Department and Education Distribution a) Department distribution b) Education distribution.

h. Gender Distribution

Figure 3.15 shows the variation of promotion percentage concerning gender. There is a major difference in percentages across genders. If we look at the gender, males have around 70% of the data and females 30%.

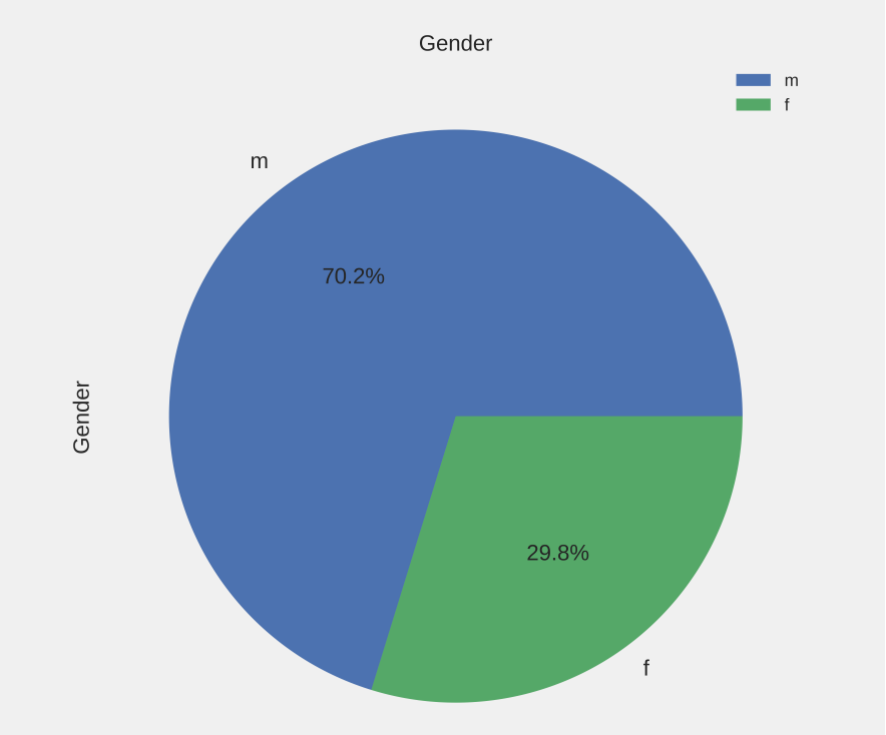


Figure 3.15 Gender Distribution.

Contrary to the assumption, females have more promotions as compared to males. Figure 3.16 show that the two genders have equal proportions of promotions. This does not mean that an equal number of females and males were promoted. As established earlier, the population of males is far greater than females. The proportions calculated are concerning their population.

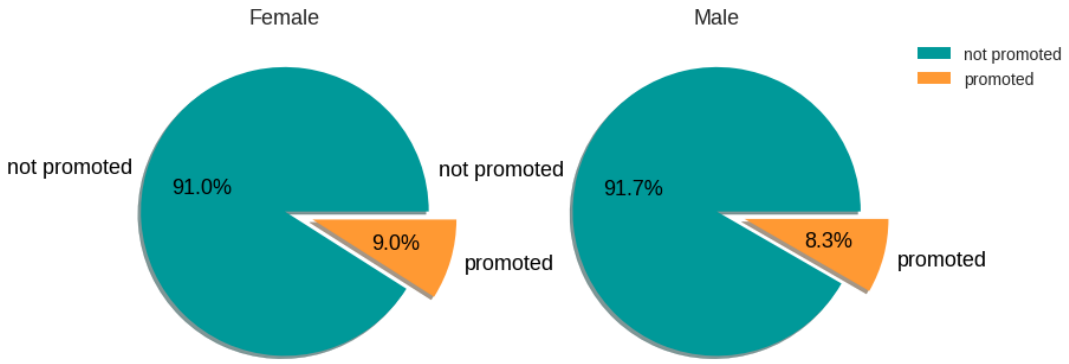
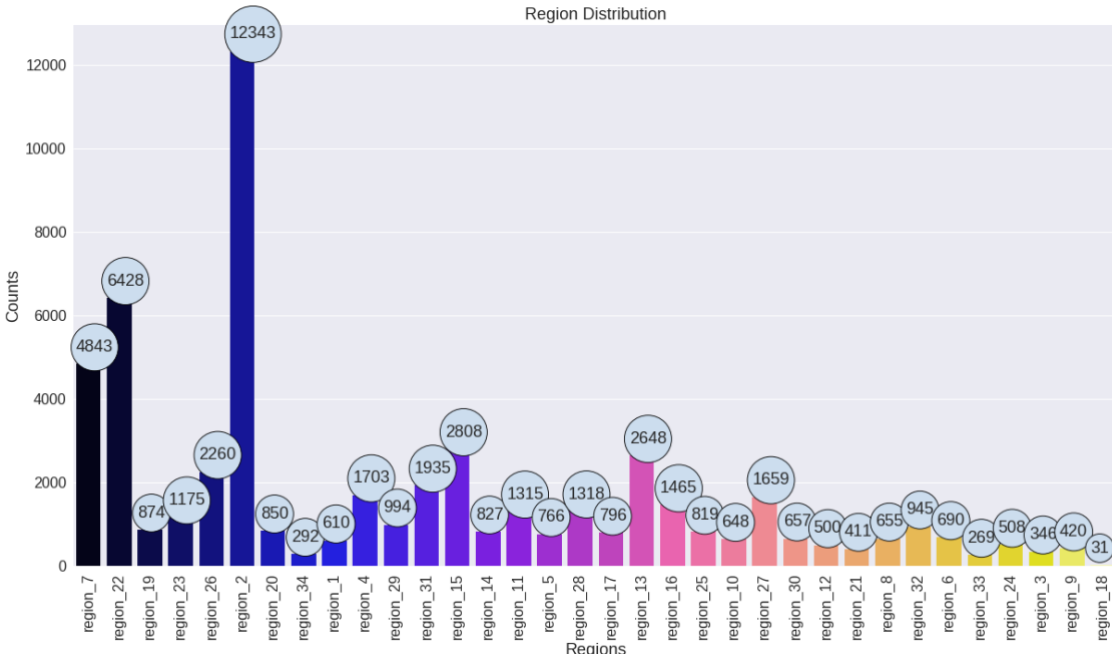


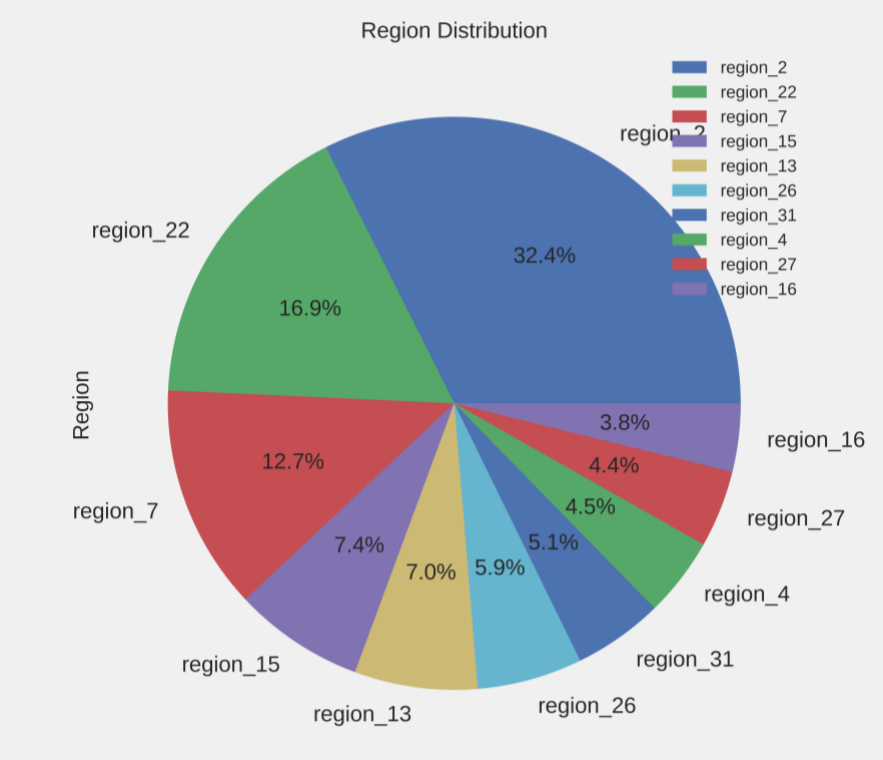
Figure 3.16 Gender Distribution with promotion.

i. Region Distribution

In Figure 3.18, when we look at the region, region_2 has 32% of the vote, region_22 has 16%, and region_7 has 12%. This means that these three regions cover almost 60% of the data.



a)



b)

Figure 3.17 Region Distribution a) in number, b) in percentage**.**

j. Recruitment Channel Distribution

Figure 3.18 demonstrates the variation of promotion percentage with the recruitment channel that they have come from. According to the data, the percentage of promotions is higher among the employees who were recruited through referrals. Referred cases account for 2% of all cases, while sourcing accounts for 42%.

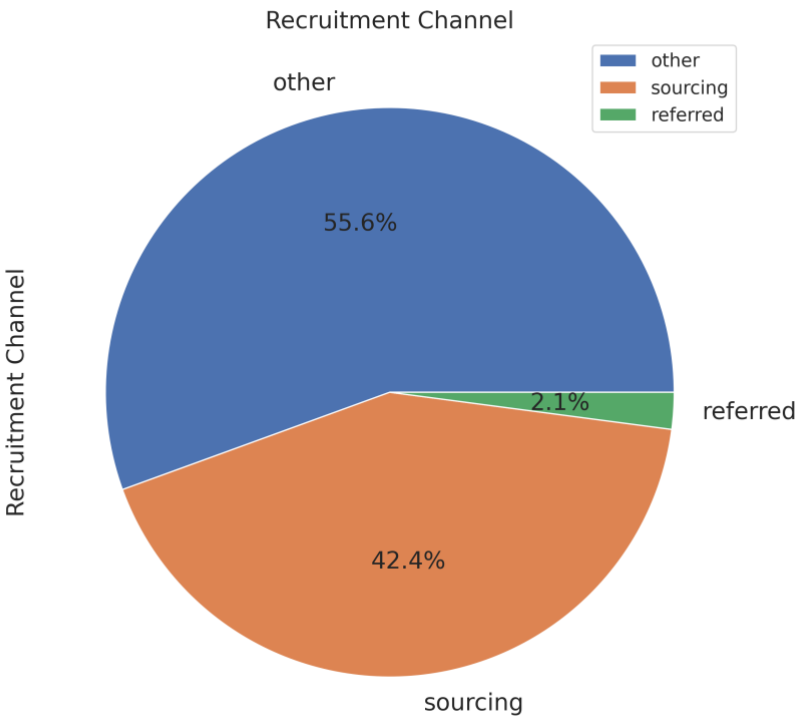


Figure 3.18 Recruitment Channel Distribution.

k. Is Promoted Distribution

From Figure 3.19, it is observed that the target class is highly imbalanced, and we must balance these classes of the target class. Using machine learning models with imbalanced classes often leads to very poor results that are completely biased towards the class having a higher distribution. When observing the data imbalance, the number of promoted people is very small when compared to non-promoted people, and this is challenging in modeling. The data is not balanced. The promoted employees number only 4668, and the non-promoted employees' number 50140; a 91% and 9% ratio which is very unbalanced.

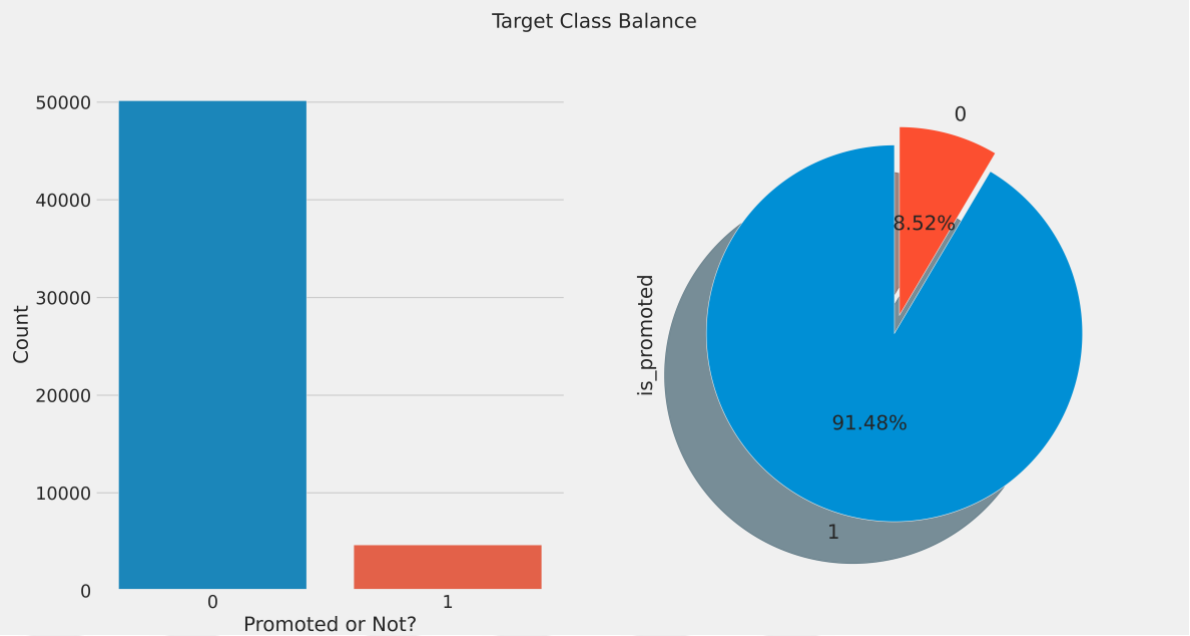


Figure 3.19 Is Promoted Distribution.

3. 2. 2 Bi-variate analysis

One of the most basic types of quantitative analysis is bivariate analysis. It serves two variables to determine the empirical link between them. Bivariate analysis can be useful in evaluating basic association hypotheses. In this section, we examine variables at a predetermined significance threshold. Bivariate analysis may be used for any grouping of absolute, categorical, and continuous variables. These are classified as Continuous & Continuous, Categorical & Categorical, and Categorical & Categorical. During the analytic process, many strategies are utilized to manage these groups. The following are the specific combinations that are possible.

A. Department & Education versus Employee Promoted

When comparing the effects of different departments and promotions, it can be seen that most of the employees in the company are from Sales, Marketing, Operations, and Procurement. These departments have more promotions, but the ratios of promotions are very high, so we can assume that competition is very high for promotions in these departments. While the Technology department had the highest percentage of employees getting promoted, the Legal department had the lowest number. There are no major differences seen in terms of percentages.

According to Figure 3.20 **Error! Reference source not found.**, practically all departments have a pretty comparable influence on promotion. As a result, we may conclude that all departments have a comparable influence on promotion. This column also comes out to be less important in making a machine learning model, as it does not contribute at all when it comes to predicting whether the employee should get a promotion.

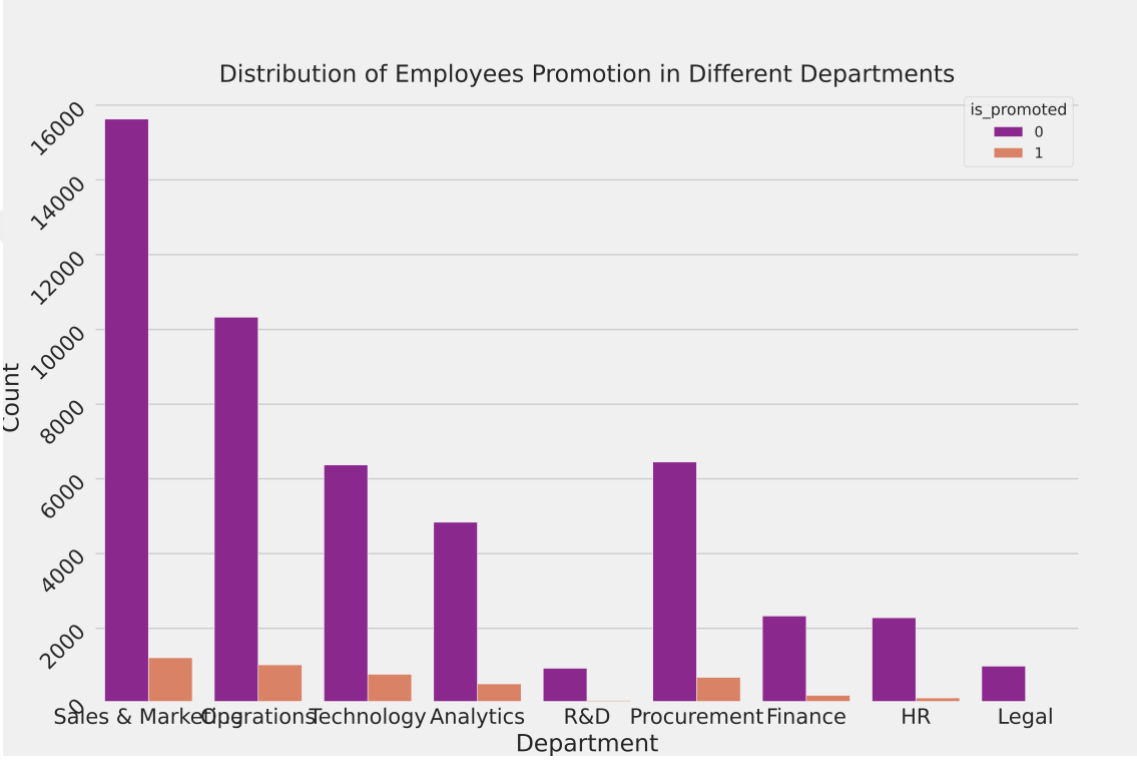


Figure 3.20 Distribution of Employees Promotion in Different Departments.

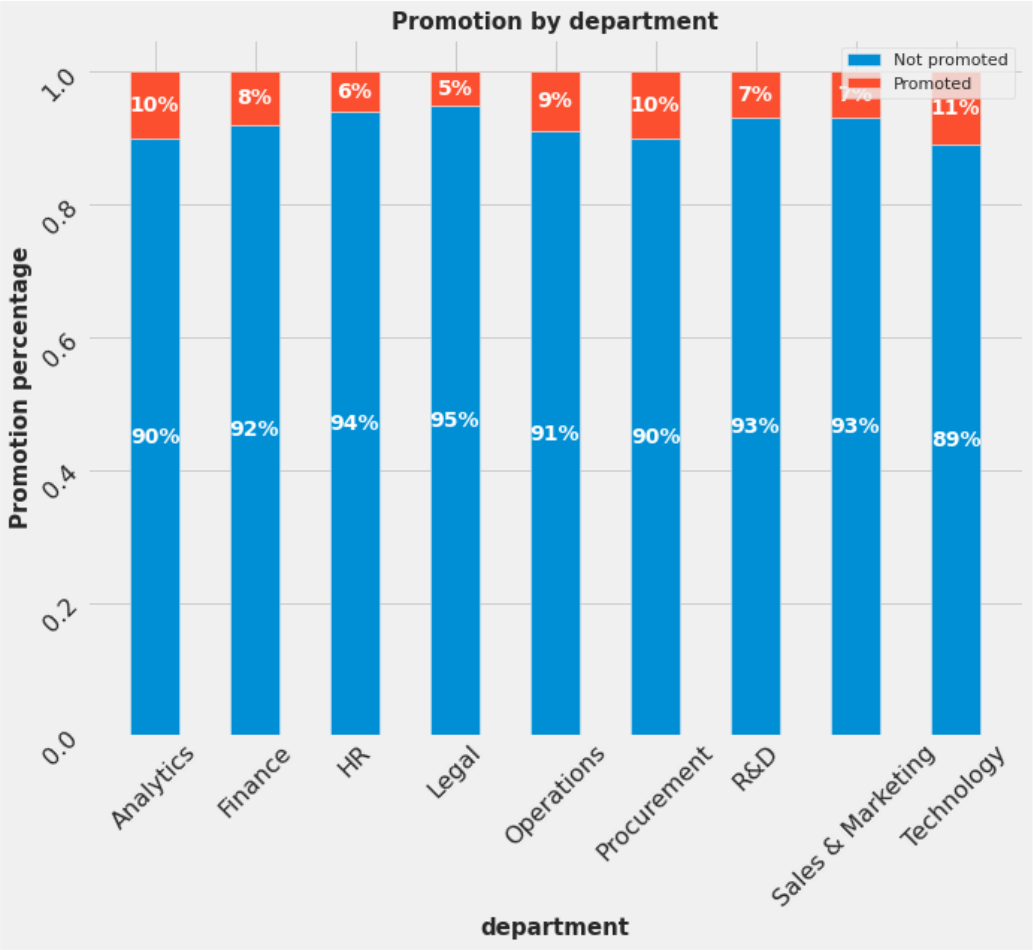
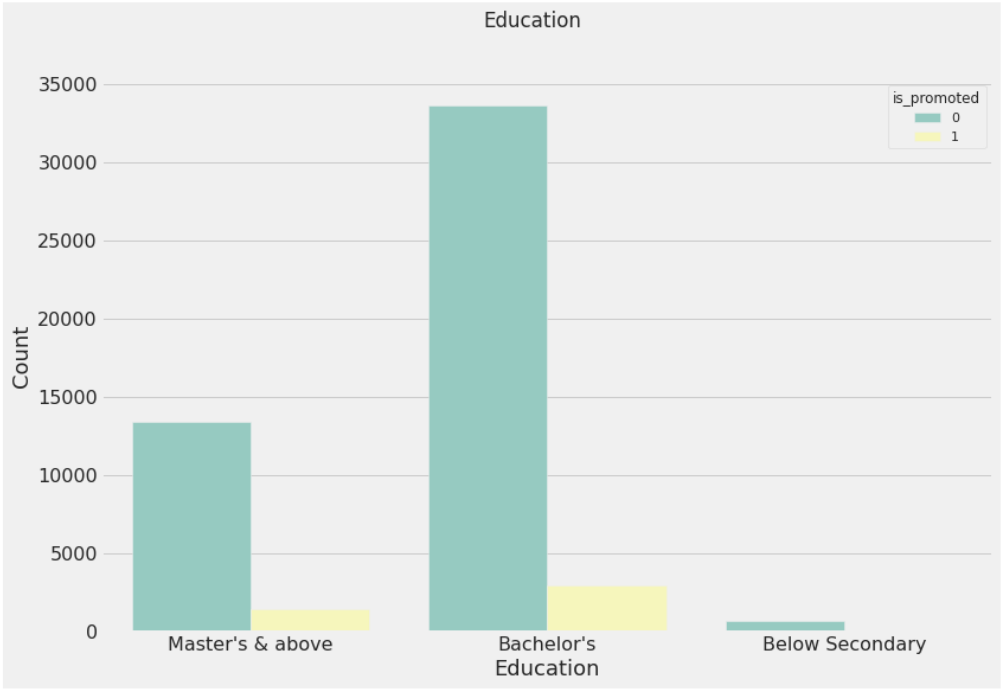
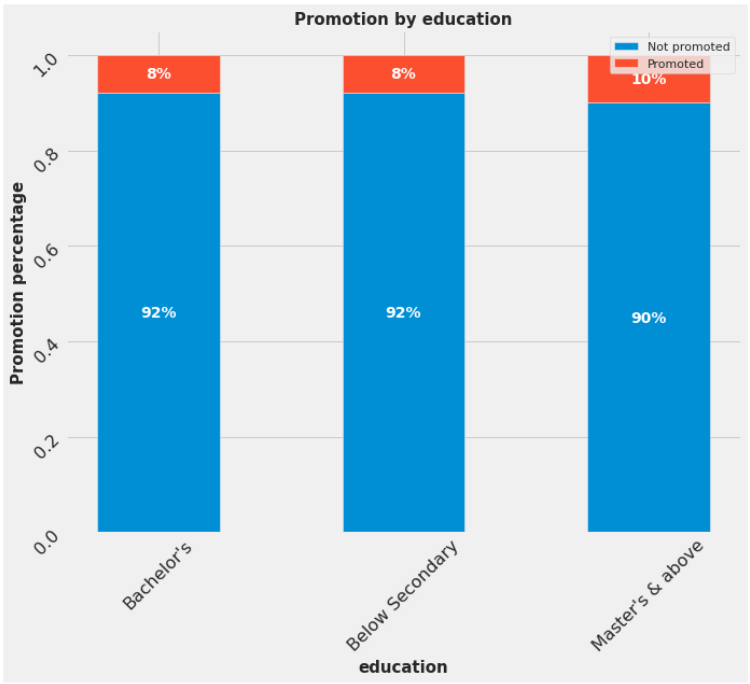


Figure 3.21 Department versus Employee Promoted.

In Figure 3.22, when comparing the percentage of promotion data by education are pretty much the same percentages across different educational backgrounds.



a)

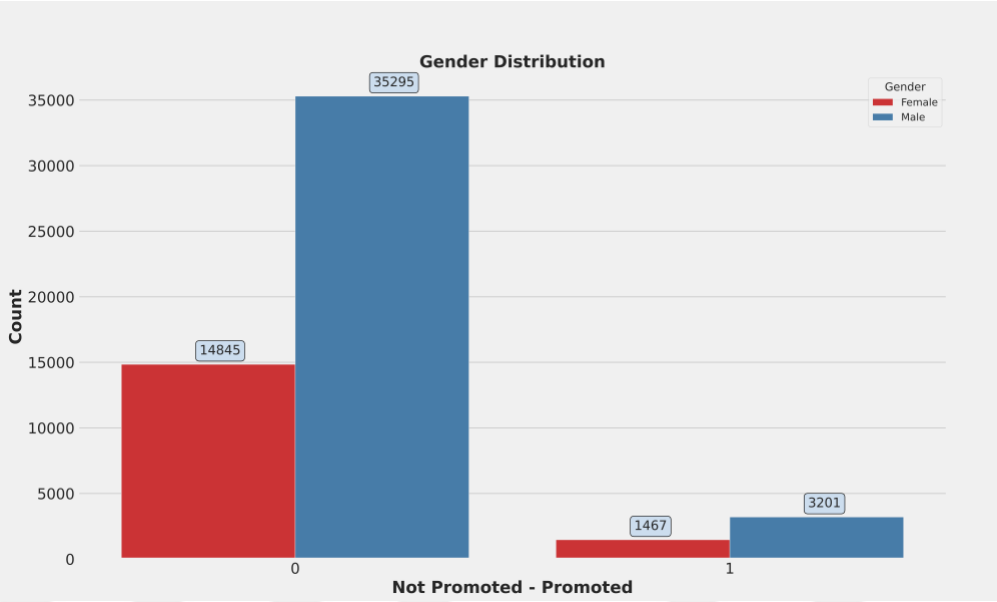


b)

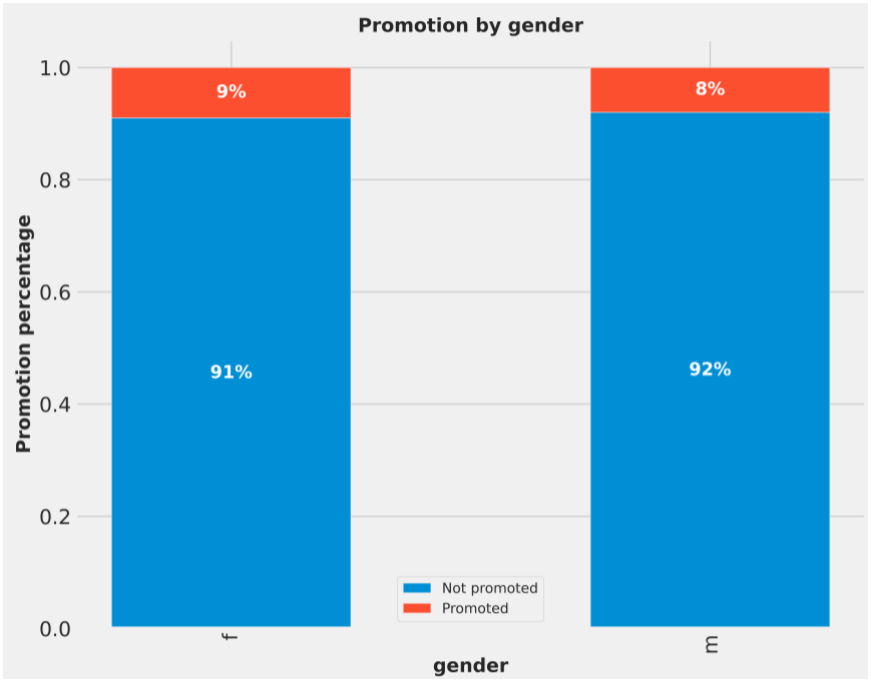
Figure 3.22 Education versus Employee Promoted a) in number, b) in percentage**.**

B. Gender versus Employee Promoted

According to Figure 3.23, male employees are promoted at a higher rate than female employees. Furthermore, male employees continue to be promoted at a higher rate than female ones. As previously stated, women are in the minority, but when it comes to promotion, they compete head-to-head with their male counterparts.



a)



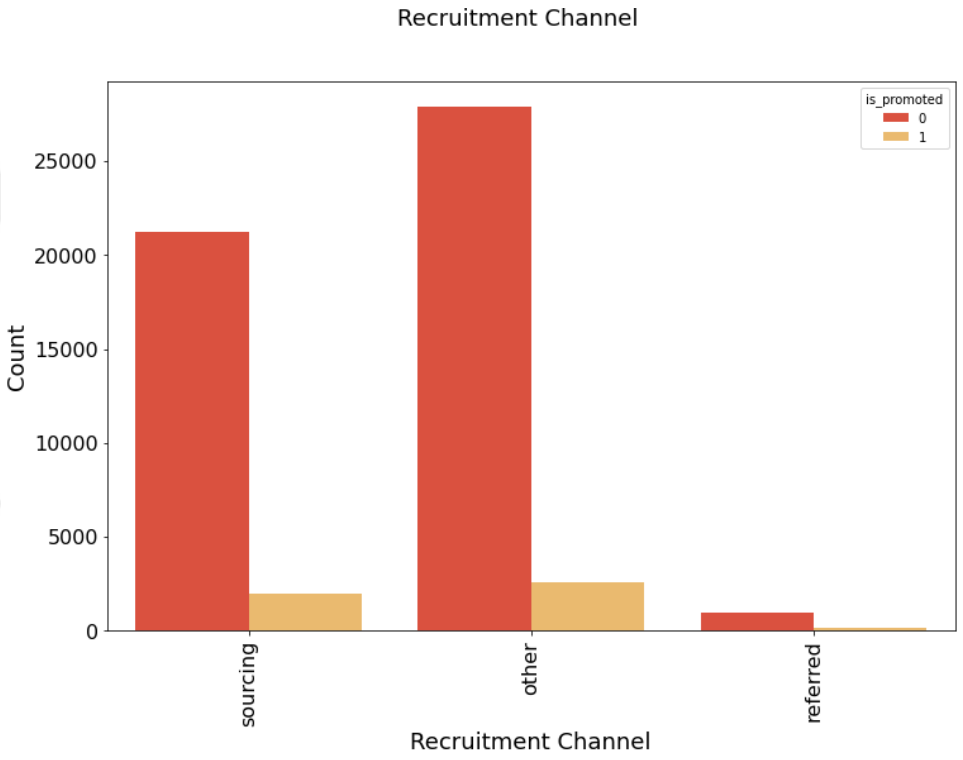
b)

Figure 3.23 Gender versus Employee Promoted a) in number, b) in percentage.

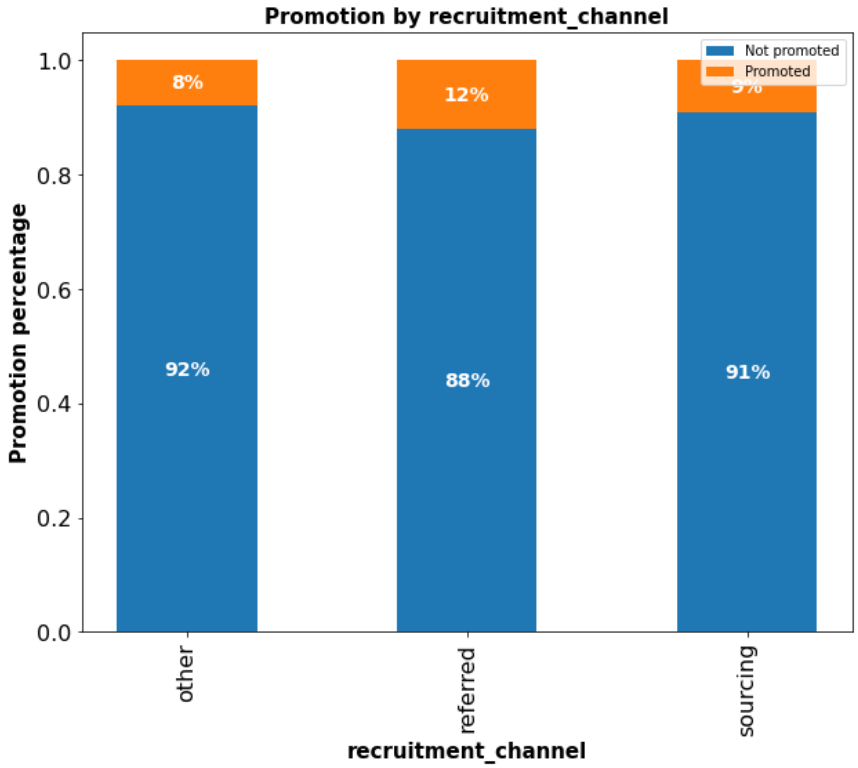
C. Recruitment Channel & Region versus Employee Promoted

From Figure 3.24 Recruitment Channel versus Employee Promoted a) in number, b) in percentage., it can be seen that the recruitment channel is not influencing the promotions. The Recruitment Channel says that the referred employees are very small,i.e.

most of the employees are recruited either by sourcing, some other recruitment agency, sources, etc.



a)



b)

Figure 3.24 Recruitment Channel versus Employee Promoted a) in number, b) in percentage.

A pattern can be observed here in Figure 3.25, employees are more concentrated in region 2, and the majority of employees are promoted from this region. Since there is more personnel in these regions, promotions are higher in regions 7, 22, and 2. Region 4 appears to have the greatest rate.

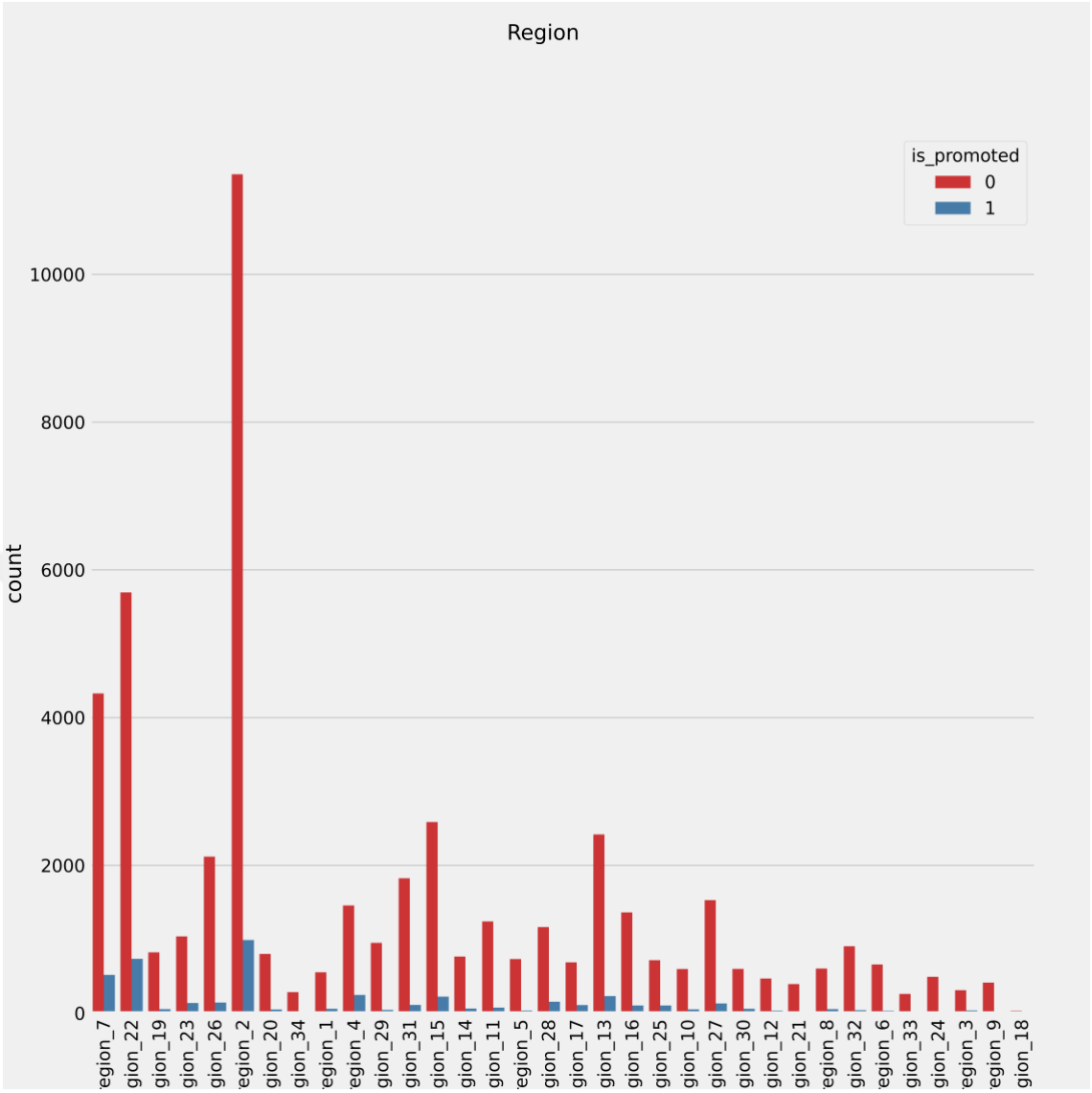


Figure 3.25 Region versus Employee Promoted.

D. Previous Year rating & No. of training versus Employee Promoted

Based on Figure 3.26 and Figure 3.27 , more employees who have attended training once were not promoted than those who were. When we are checking the distribution of training undertaken by the employees, it is visible that 80% of the employees have taken the training only once, and there is a negligible number of employees who have attended training more than three times. Moreover, employees who have had prior experience of 3 or 5 years have a higher starting point, and they also have a higher number of

promotions. Based on the bar plot above, we can see that employees who get ratings of 3 and 5 are more likely to be promoted.

Figure 3.26 No. of Training versus Employee Promoted.

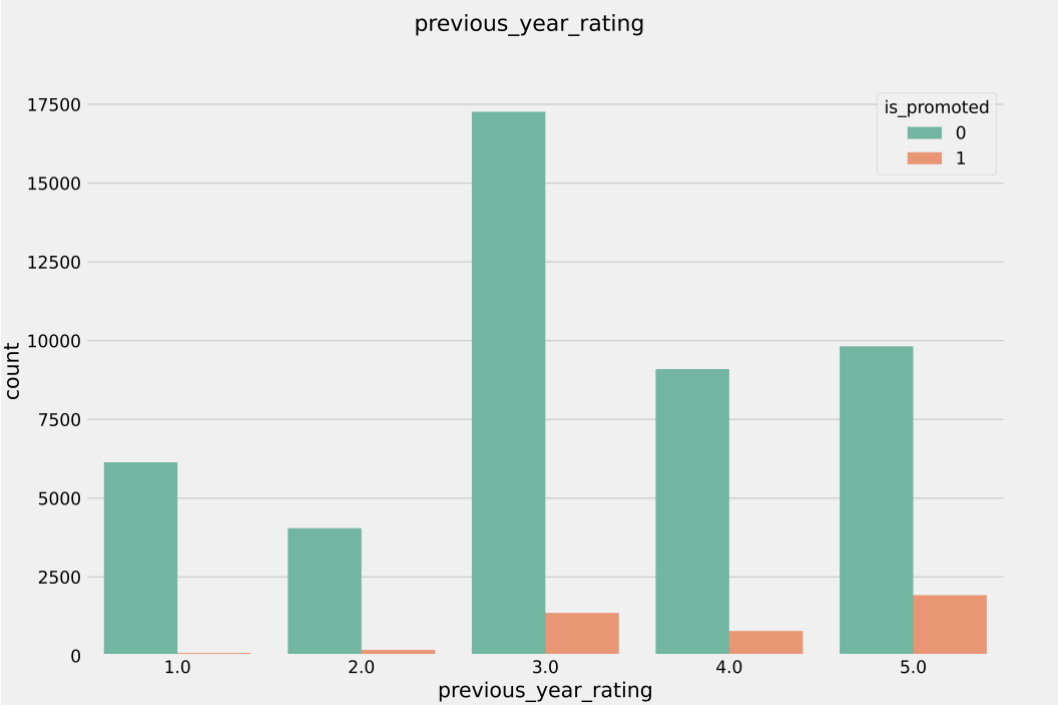


Figure 3.27 Previous Year Rating versus Employee Promoted.

E. Avg training score

In Figure 3.28, promotions follow the same pattern as the average training score, making it difficult to forecast whether a person will be promoted or not based on a certain average score. The promotion ratio increases with the score, and the ratio is very high in the 90-100 range, which means getting promoted is highly dependent on the average score. Furthermore, the percentage of promotions with an average score of 90 or higher is quite high; everyone with an average score of 90 or higher has received a promotion.

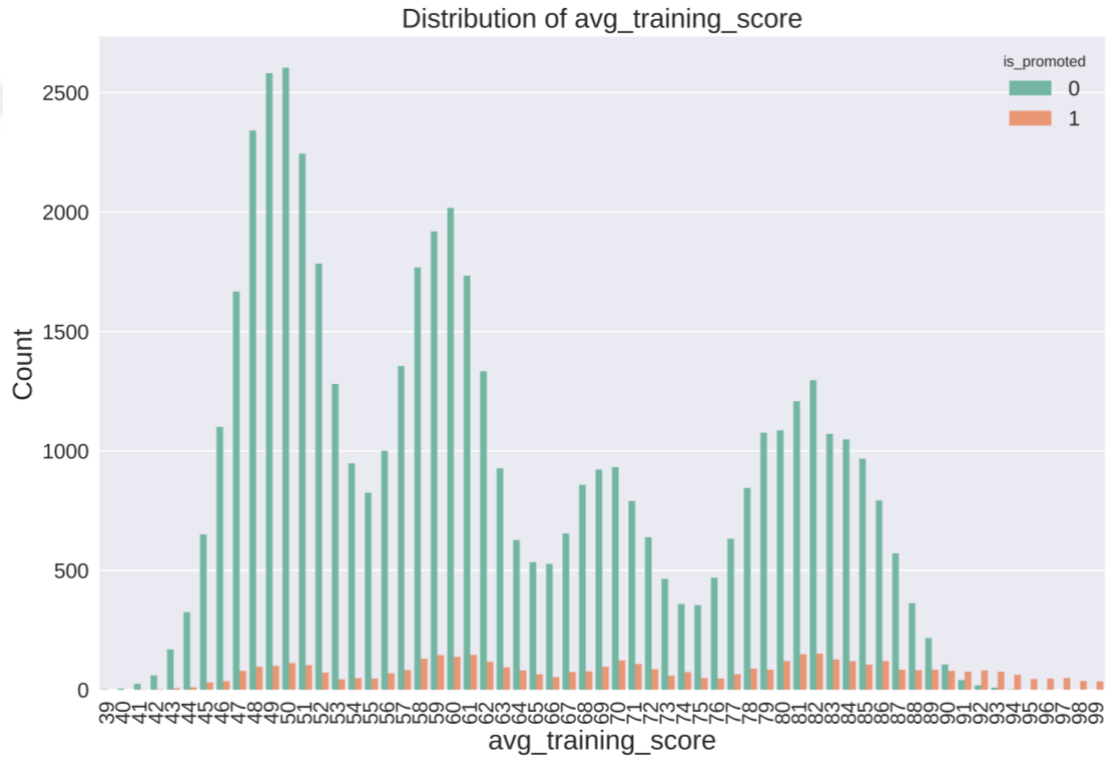


Figure 3.28 Avg training score versus Employee Promoted.

3. 2. 3 Multivariate analysis

Multivariate analysis is based on multivariate statistics concepts, which entail the observation and analysis of several statistical result variables at the same time. First, we will examine the association between the numerical columns using the Correlation Heatmap.

The heatmap is used to display the correlation between the columns, which is highly beneficial for regression issues because one of the assumptions of the linear model is that the features should not correlate. Here from Figure 3.29, we can see some obvious results, such as length of service and age are highly associated. It can also be seen that KPIs and previous year's ratings are correlated to some degree, implying that there is some relationship However, to avoid multicollinearity, we will do some feature engineering before modeling. Furthermore, in the modeling phase, a comparison of results will be done after removing those two variables to see if it is really necessary to keep or remove those features.

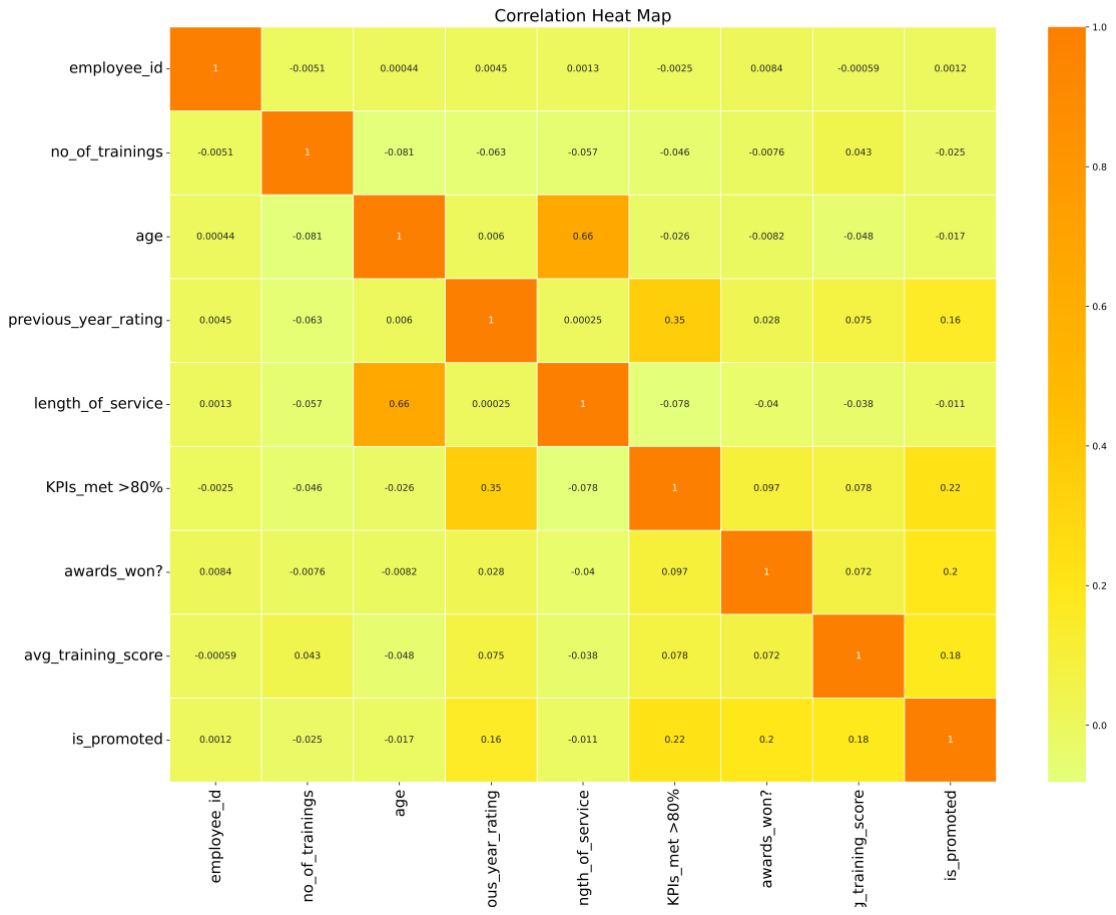


Figure 3.29 Correlation Heat map.

3. 2. 4 Data visualizations

In this part, data visualization is conducted on continuous and categorical variables to understand the link between these variables and our objective variables. Graphs are the most effective way to comprehend the behavior of characteristics and the relationships between them.

a. Education with Department

From Figure 3.30, the Education Level versus Department, we observe that most of the employees are Bachelor's degree holders. Surprisingly, most of the employees in the sales and marketing departments are Bachelor's degree holders as well.

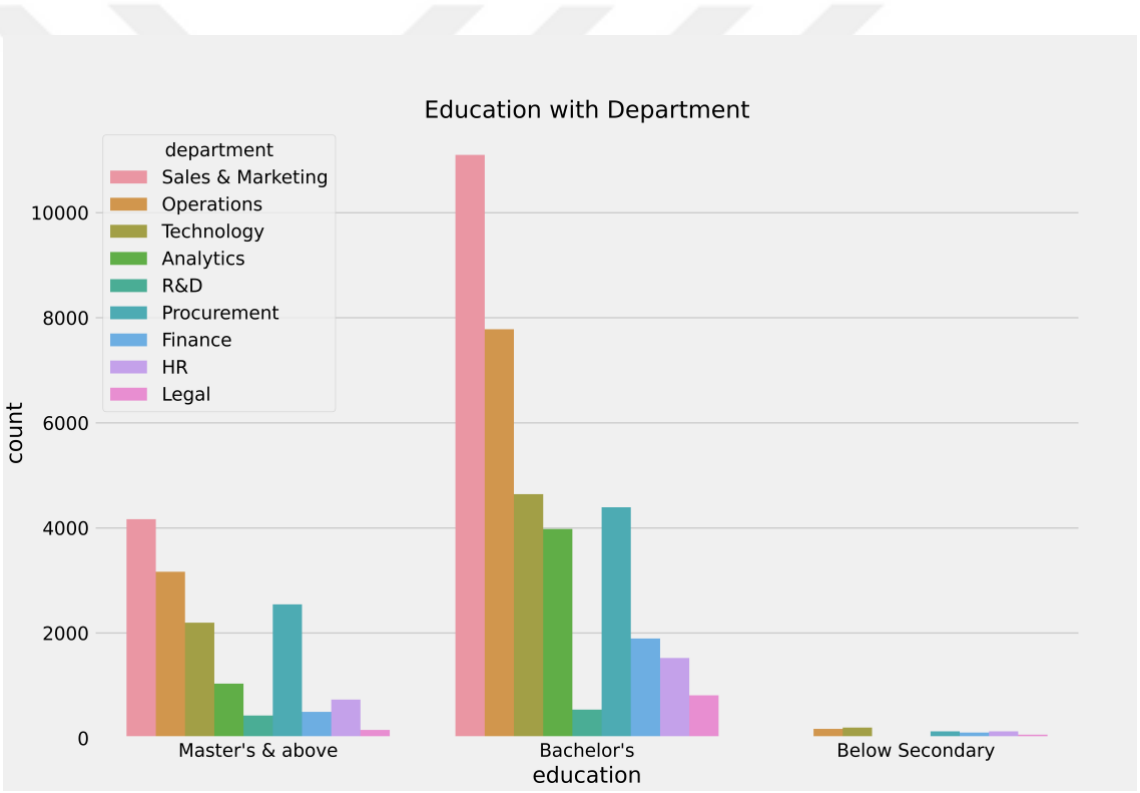


Figure 3.30 Education with Department.

b. Recruitment Channel with Department

The Recruitment Channel with the Department is shown in Figure 3.31. Many employees are from other recruitment channels, some are from sourcing, and referred employees are very rare. For the Department, it can be seen that most of the employees in the company

are from sales, marketing, operations, and procurement, and these departments have more promotions.

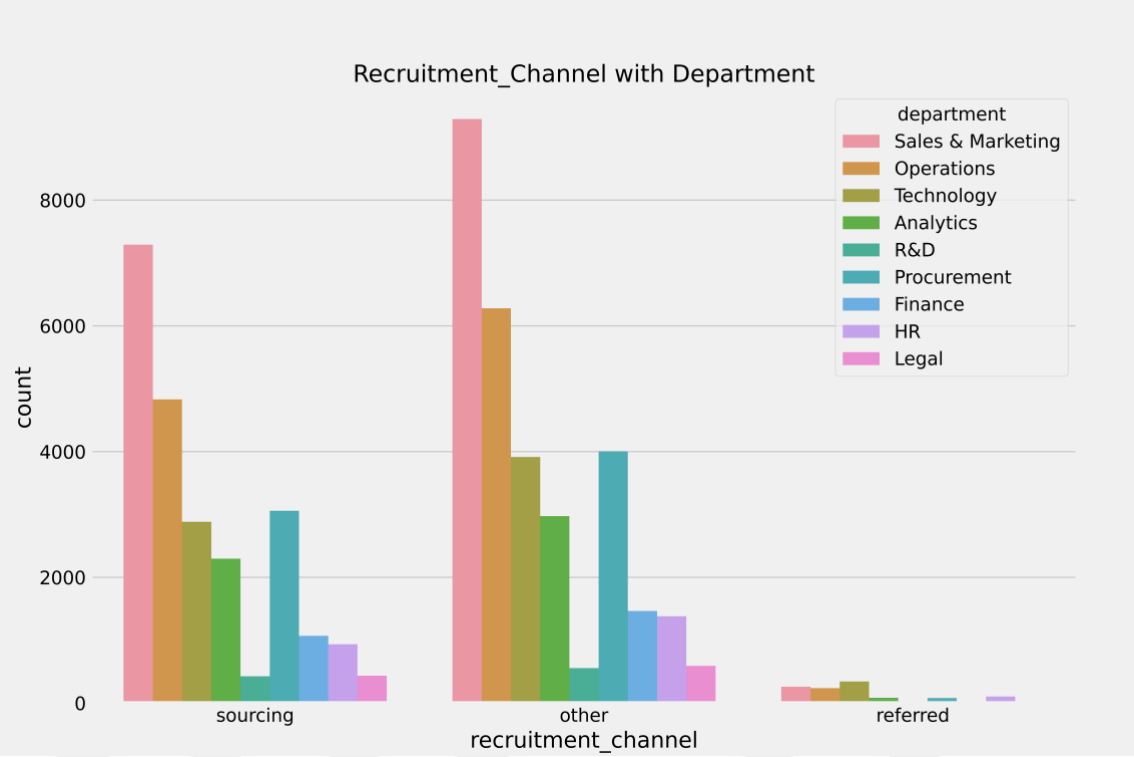


Figure 3.31 Recruitment Channel with Department.

c. Gender with Department and Promotion

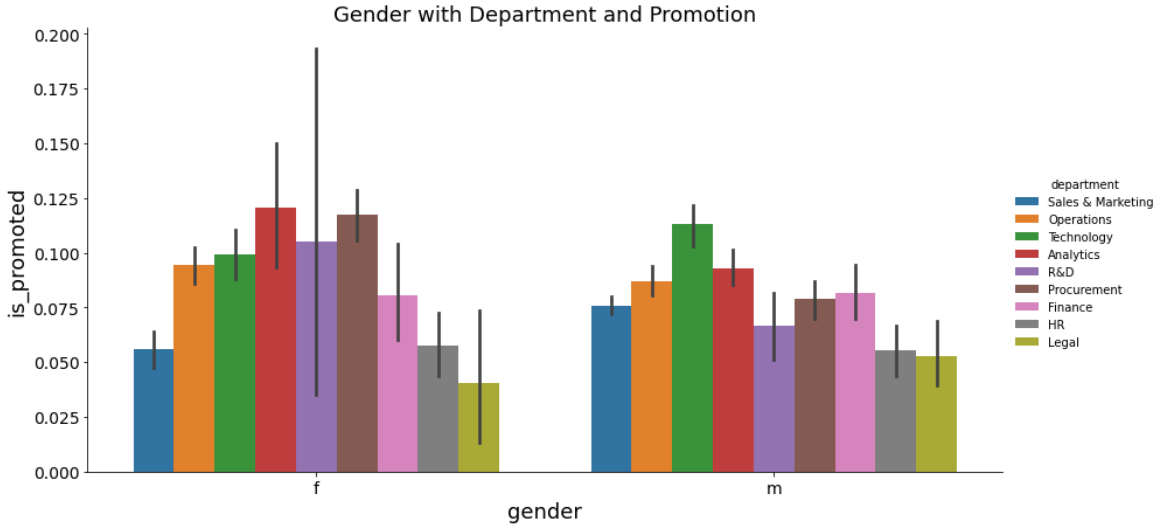


Figure 3.32 shows the gender breakdown by department and promotion. This figure represents the distribution of females and males in the department section. There are

varying percentages in the departments, and for females, there is a higher promotion in the two departments of analytics and procurement. Unlike males, the two sections with the highest percentages are technology and analytics.

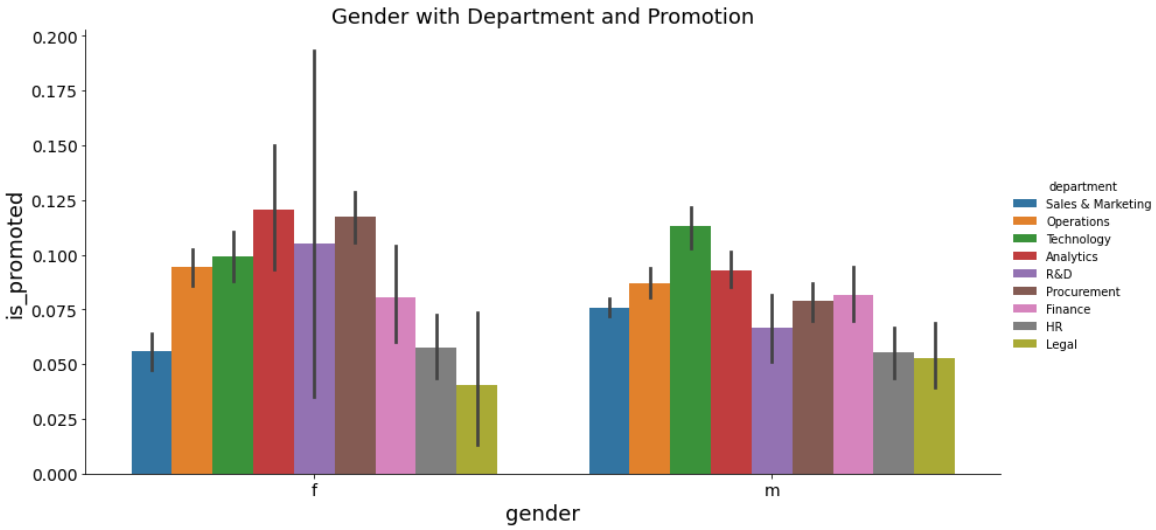


Figure 3.32 Gender with Department and Promotion.

d. Gender with Recruitment Channel and Promotion

Gender with Recruitment Channel and Promotion is shown in Figure 3.33. It shows that most of the promoted employees, both females, and males, are recruited by referral, then sourcing, and others almost always fall into the same percentage.

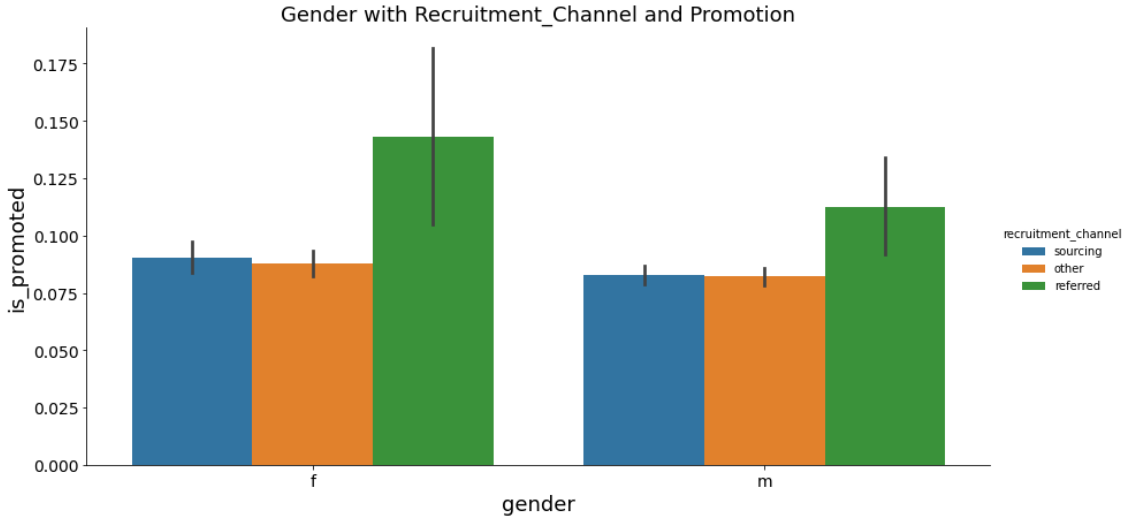


Figure 3.33 Gender with Recruitment Channel and Promotion.

e. Department with Recruitment Channel and Promotion

Figure 3.34 shows the distribution of departments with recruitment channels and promotion. From this figure, it can be concluded that the majority of promoted employees in all departments are recruited by referral, and the other recruitment channels are almost equal in proportion.

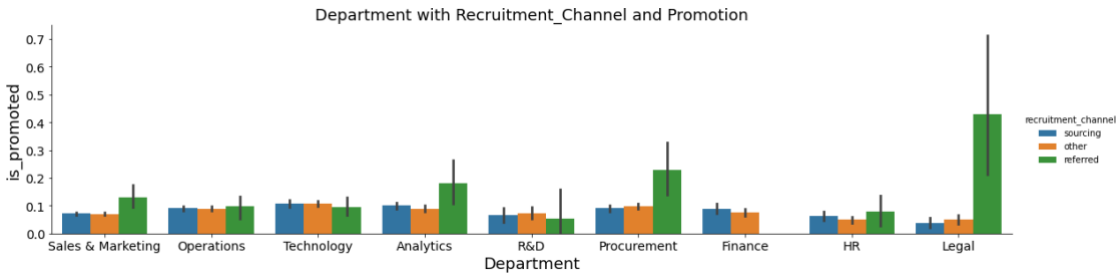


Figure 3.34 Department with Recruitment Channel and Promotion.

f. The relationship between Departments and Promotions when they won awards

The relationship between departments and promotions when they win awards is shown in Figure 3.35. It is observed that females and males have the same chances of promotion in all departments and that technology analytics and R&D departments have the highest chance of promotion when they win awards.

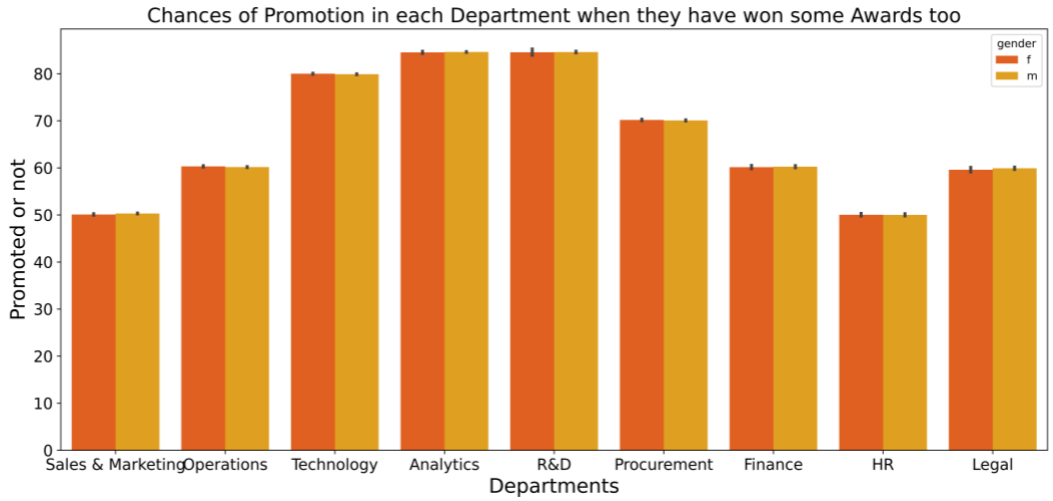


Figure 3.35 The relation between Departments and Promotions when they won awards.

3. 3 Machine Learning Algorithms

Machine learning (ML) is a branch of computer science and artificial intelligence with significant links to statistics and optimization. It is about learning from data rather than following strictly explicit programming instructions. A machine learning framework begins with information preparation (training) by extracting knowledge from training data and then employs that trained knowledge to anticipate the output of fresh data (testing). Machine learning can be classified as supervised, unsupervised, semi-supervised, or reinforcement learning. These algorithms are grouped depending on the expected output of the algorithm or the type of input accessible during machine training. In this thesis, we use supervised learning approaches for binary classification. The training set in supervised learning (Saradhi and Palshikar 2011), comprises labeled data, and the system infers a function mapping from inputs (usually vectors) to outputs (labels). The algorithm employs this inferred function from the training process to categorize fresh data during the testing step.

XGBoost, Random Forest, Decision Tree, Logistic Regression, AdaBoost, and Gradient Boosting are the different types of supervised algorithms applied in the classification phase of our study. These algorithms are used through the Scikit-learn library and the experiment is carried out within Python.

1. XGBoost

XGBoost is a boosted tree approach based on the gradient boosting principle. XGBoost employs more precise approximations by applying second-order gradients and enhanced regularization. When compared to others, it uses a more regularized-model reinforcement to control overfitting and hence improves performance. It is a fast approach based on parallel tree creation that is designed to be fault resistant in a distributed situation (Jain and Nayyar 2018). The classifier accepts data in the form of DMatrix (Saradhi and Palshikar 2011). It is considered an internal data structure employed by XGBoost for memory efficiency and speed enhancement. During the research, the following characteristics were investigated and incorporated:

Firstly, regularization; is the primary advantage of XGBoost. Because standard Glioblastoma (GBM) implementations lack regularization like XGBoost, it also aids in

reducing overfitting. The proposed method is a technique used in linear and tree-based models to prevent overfitting. Secondly, parallel processing; XGBoost uses this and is much quicker than GBM. XGBoost now supports the Hadoop implementation. Users can define custom optimization targets and assessment criteria in XGBoost. This gives a completely new dimension to the model, and there are no restrictions on what we may accomplish. In addition, XGBoost has a procedure for dealing with missing values. The user must offer a value that differs from the other observations and pass it as a parameter. As it encounters a missing value on each node, XGBoost tries different things and learns which path to follow for missing values in the future.

Furthermore, when a GBM encounters a negative loss in the split, it will cease dividing the node. In other words, it is a 'greedier' algorithm. On the other hand, XGBoost divides up to the max depth set before pruning the tree backward and removing splits beyond which no positive benefit is obtained. Another benefit is that a division of a negative loss, say -2, can be followed by a division of a positive loss, say +10.0, on occasion. GBM would come to a halt when it reaches -2. XGBoost, on the other hand, will dig deeper and observe a combined impact of +8 of the splits and maintain both. Therefore, cross-validation is supported by XGBoost at each iteration of the boosting process, making it straightforward to acquire the precise optimal number of boosting rounds in a single run. In contrast to GBM, we must execute a grid search and only a restricted number of parameters may be examined. Finally, the user can begin training an XGBoost model from the previous run's last iteration. This can be a substantial benefit in some situations. This capability is also available in the GBM implementation of sklearn, thus they are on the same page (Aarshay 2020).

2. Random Forest (RF)

Random Forest is a classifier that uses several decision trees on different subgroups of a given dataset and averages them to enhance the predicted accuracy of that dataset. Rather than depending on a single decision tree, the random forest collects forecasts from each tree and predicts the ultimate output based on the majority of votes cast on the predictions. The larger the number of trees in the forest, the higher the accuracy and the lower the risk of overfitting. In addition, the RF method is an ensemble learning strategy for classifying and backsliding the dataset. This method works when outputting the mode of the classes

(categorizing) or when backsliding the specific tree by creating a huge number of DT (Jaiswal, n.d.).

3. Decision Tree (DT)

A decision tree is a flow that is used to traverse all available alternatives and their outcomes. Each "branch" indicates a possible option when making a decision. Decision trees may be indefinitely scaled and are based on cause and effect. When a consequence leads to a different line of action, we may extend that branch, and so on. A decision tree chart may help us examine alternatives and their outcomes before committing to a solution, allowing us to make the best decision with the least amount of harm and the greatest benefit. It provides a stylized universe in which we may play out a sequence of actions and see where they go without devoting unnecessary real-world time and resources. It is termed a decision tree because, like a tree, it begins with the root node and develops on subsequent branches to form a tree-like structure. It also breaks the decisions so that they may be shown graphically and clearly (Jaiswal, n.d.).

4. Logistic Regression (LR)

Logistic regression is an important machine learning technique because it can offer probabilities and classify new data using continuous and discrete datasets, and seeks to calculate the likelihood that the output variable belongs to a certain class. Logistic regression may be used to categorize observations using many forms of data and can quickly discover the most efficient factors for classification (Jaiswal, n.d.).

5. AdaBoost

AdaBoost, also defined as Adaptive Boosting, is a classifier that uses ensemble boosting. It combines many classifiers to improve classifier accuracy. AdaBoost is a technique for generating iterative ensembles. The AdaBoost classifier creates a strong classifier by merging numerous low-performing classifiers, resulting in a high-accuracy strong classifier. The main assumption of AdaBoost is to build classifier weights and train the data sample in each iteration to assure accurate predictions of unexpected events. AdaBoost must fulfill two requirements (Navlani, n.d.):

the classifier should be interactively trained on a variety of weighted training instances,

and

it must seek to offer a good match for these instances in each iteration by reducing training error (Navlani, n.d.).

6. Gradient Boosting

One of the most successful machines learning algorithms is the gradient boosting strategy. Machine learning algorithm mistakes are broadly categorized into two types: bias errors and variance errors. As one of the boosting strategies, gradient boosting is used to decrease the bias error of the model. Gradient boosting requires each prediction to outperform its previous by minimizing errors. Its distinguishing feature is that, rather than fitting a predictor to the data at each iteration, it instead fits a new predictor to the residual errors created by the preceding prediction (Tarbani 2021).

4. METHODOLOGY

Our methodology is mainly composed of five phases; Input Data (which includes Data Understanding & Visualizing), Data Pre-processing (which includes Data Cleaning, Data Preparing & Data Splitting), Data Manipulation (which includes Data Preprocessing and Manipulation), Data Modeling, and Data Evaluation (Fine & Tune). These phases are described as follows:

- Input Data: The Data Understanding & Visualizing phase includes conducting an Exploratory Data Analysis (Univariate, Bivariate, Multivariate). The data will be described using statistical and graphical approaches. During the data exploration phase, the methodologies of variable identification, univariate analysis, bivariate analysis, and multivariate analysis will be applied to the Analytics Vidhya dataset step-by-step (Figure 4.1).
- Data Pre-processing: The clean, prepare, and split data phase includes many steps, such as data imputation of invalid or missing data, and fixing column names. The Permutation Feature Importance was used to select the appropriate variables, also forming some new features, which is Feature Engineering. To choose the most significant features, feature selection was used to remove irrelevant features. Furthermore, the dataset is divided into two parts: training and testing. Training data is used to train the model, while testing data is used to test the model. The data were divided into training (80%) and testing (20%). This phase aims to eliminate imperfect information and implement good engineering features to have suitable data for our problem (Figure 4.1).
- Data Manipulation: Preprocessing and Manipulating Data is another important phase of our study. Earlier, in this problem, we noticed that the target column is highly imbalanced. We need to balance the data by using some statistical methods. Here, the Synthetic Minority Oversampling Technique (SMOTE) method is used to oversample our data. Feature scaling is another method used to normalize the range of independent variables or features of data (Figure 4.1).
- Data Modeling: The modeling phase is the whole process of training and testing components. To construct our model, we trained it on a training set and validated it on a test set. We ran all of our analyses on the training set and validated them

on the testing set. XGBoost, Random Forest, Decision Tree, Logistic Regression, AdaBoost, and Gradient Boosting machine algorithms are used and tested as classification algorithms for the prediction model. The best model is used to test various classifiers (Figure 4.1).

- Data Evaluation: In the last phase, the evaluation (fine and tune) phase, every occurrence of the above situation would be categorized based on whether or not the employee promotes the firm. The number of cases properly categorized by the model may be determined using a typical confusion matrix. A classification report would show the model's accuracy, precision, recall, and F1-Score. The classifiers are assessed using the evaluation metrics given above in order to discover the best model for the problem. Therefore, we will fine-tune the model by applying Hyperparameter Tuning and Cross-Validation iteratively until we find the best model (Figure 4.1).

Figure 4.1**Error! Reference source not found.** depicts a high-level overview of the framework that can apply to different cases. The following sections go through the specifics of each step. At the outset, the dataset will be described.

To give some brief explanations about the implementation of our study, algorithms are used through the Scikit-learn library, and the experiment is carried out within Python.

4. 1 Input Data: Data Understanding & Visualizing

For our thesis, we use publicly available data provided by Analytics Vidhya Data Analysis. This dataset has 14 characteristics and 54808 records for train data and 23490 records for test data. Not all the 14 features are taken into account in our work for employees' predictions of promotions. We will choose the relevant aspects and add new ones that impact the employee's promotion as an important indication of the promotion process.

More details of the dataset were explained in (3. 1 Data Description). Further details of the dataset can be found while explaining data preprocessing steps in the following section.

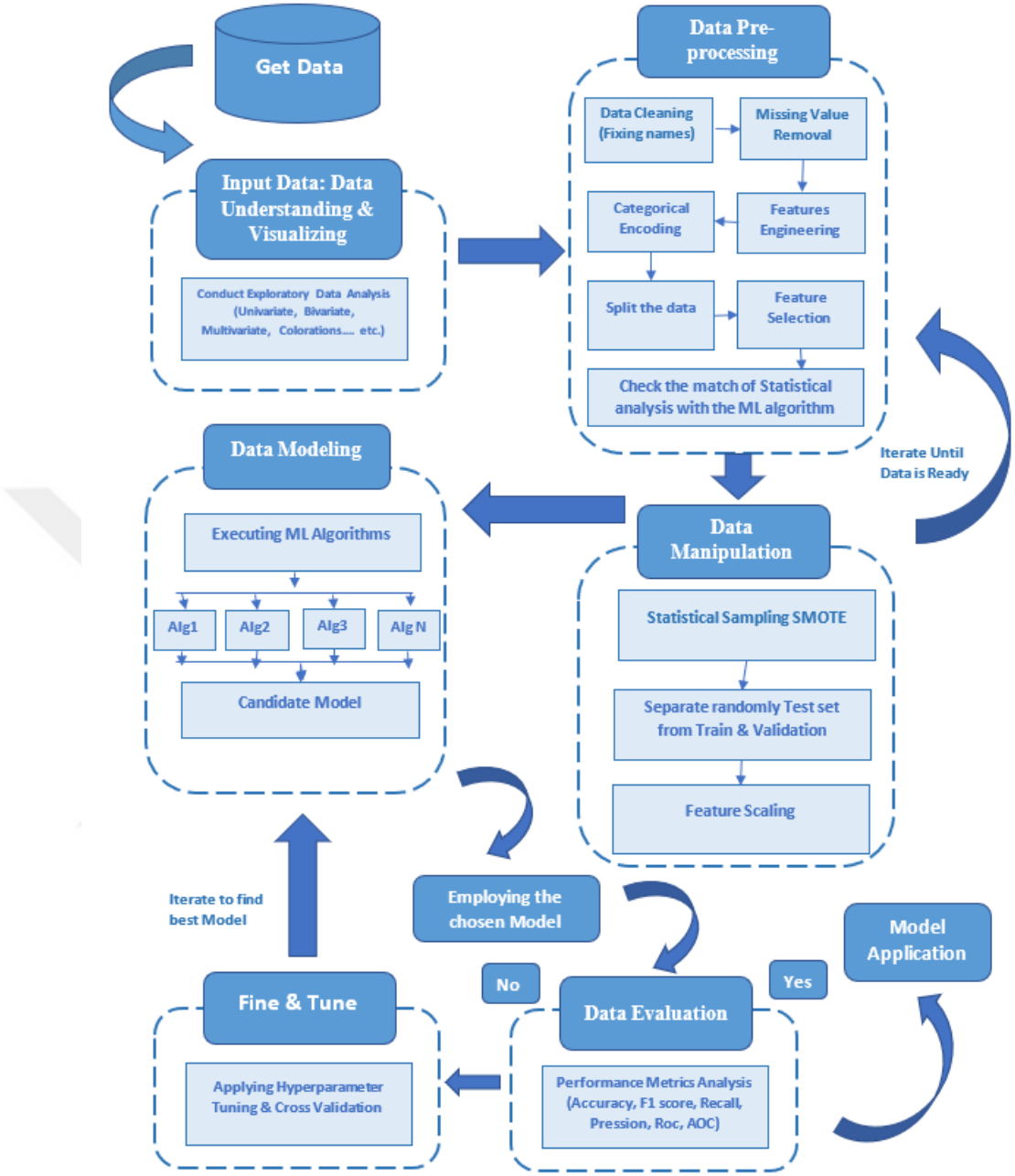


Figure 4.1 The general structure of the proposed employee promotion prediction framework.

4. 2 Data Pre-processing: Data Cleaning, Data Preparing & Data Splitting

There were instances in the raw data that were not appropriate. This was due to mistakes and abnormalities that had to be removed. The data types were then evaluated and

changed. Before applying feature selection to the dataset to identify the key characteristics and acquire a meaningful subset of key attributes to be utilized in the classification exercise, data cleaning and filling-in of missing values in the dataset were conducted. For the reasons stated above, the data preparation stage is critical for our investigation to acquire clean and usable data. Analysis of the dataset is critical at this time. The data cleaning, preparing, and splitting data are discussed in sections in this part, along with some analytical findings.

Before we do the feature engineering, we first check if there are any duplicate employee IDs, and as a result, there are not any. Following that, we change and fix the column names for awards_won? and KPIs_met >80% to awards_won and KPIs_met >80 in sequence. In addition, we removed the column employee id from our data due to its unusability.

The absence of data in the training data set reduces the capacity to fit a model or leads to a biased model since we have not thoroughly studied the functioning and connection with other variables, which can lead to inaccurate predictions. Therefore, we will need to handle column types and missing values in data cleaning.

Treatment of Missing Values is a very important step in any machine learning model creation and in our framework, as machine learning predictive models cannot work with missing values. Missing values can be caused for various reasons, such as incomplete forms, unavailable values, data entry errors, and data loss. There are many types of missing values such as random, missing values that are not random, and missing values that are completely random. Hence, to impute and treat missing values to make a good machine learning model, we can use different methods such as business logic to impute the missing values or statistical methods such as mean, median, and mode. ML techniques can be used to impute the missing values. It should also be deleted when the percentage of missing values is very high.

In Table 4.1, only two columns have missing values in the training dataset. Also, the percentage of missing values is around 4% and 7% in education and previous_year_rating, respectively. We, therefore, do not have to delete any missing values; we can simply impute the values using mean, median, and mode values. Using

the mode values, we imputed the missing values. Even for the previous year's rating, it only seems to be numerical, but in reality, it is also categorical. After importing the missing values into the training and testing datasets, we can see that there are no null values left in any of the datasets, thereby dealing with the missing data.

Table 4.1 Total missing values.

Features	**Train_Total**	**Train_Percent %**
KPIs_met >80 %	0	0
age	0	0
avg_training_score	0	0
awards_won	0	0
department	0	0
education	2409	4.4
gender	0	0
is_promoted	0	0
length_of_service	0	0
no_of_trainings	0	0
previous_year_rating	4124	7.52
recruitment_channel	0	0
region	0	0

The process of extracting features from raw data using domain expertise and data mining tools is known as "feature engineering". These factors can be used to improve the performance of machine learning algorithms. Feature engineering may be thought of as applied machine learning. There are several approaches to feature engineering. Many in the industry believe it to be the most crucial step in improving model performance. It is critical to thoroughly analyze the columns to create new features from current ones. There are many methods to perform feature engineering, such as removing unnecessary columns. We can also do it by extracting features from the date and time features or by extracting features from the categorical features. We can also do it by binning the numerical and categorical features and by aggregating multiple features together by using simple arithmetic operations. In our model, we are going to perform feature engineering by removing unnecessary columns, binning the numerical and categorical features, and aggregating some features together.

4. 2. 1 Aggregating Multiple Features

The variables need to be categorized so that the impact of making groupings can be seen more clearly since many of the variables are either continuous or have a large number of discrete values that peak at specific places. While doing these modifications, we make certain that we also perform the same for our Testing set. New features—such as Metric of sum, Total score, Work fraction, Work start year, Years remaining to retire, and Performance—are calculated based on the following assumptions:

- **Metric of sum:** this feature is the sum of awards won, KPIs met, and the previous year's rating.

  Figure 4.2 demonstrates employees having a metric of sums 4,5 and 6 have been promoted, unlike those who have sums 1 and 2, who have a low chance to be promoted.

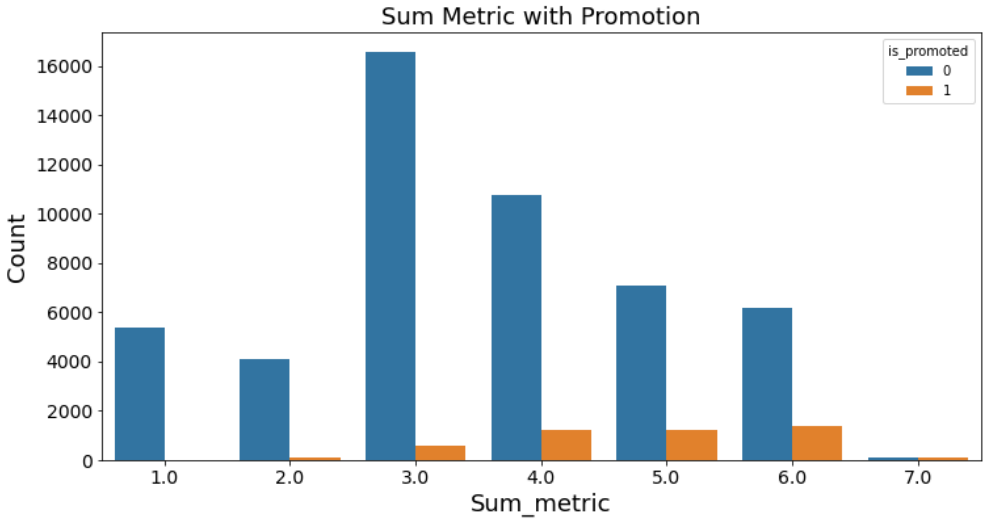


Figure 4.2 Sum_metric distribution.

- **Total score:**

  The columns "number of trainings" and "average training score" describe the number of company-organized workshops and trainings that the employee attended, as well as the average training score for those trainings. Training and workshops are essential for employees since they are conducted to help employees grow their skills. These training ratings assist the organization in

determining whether staff are progressing. Because the two columns cannot be compared amongst employees, they do not offer a good assessment.

Assume employee A has an average training score of 60 but has only attended one workshop, whereas employee B has an average training score of 50 but has attended three workshops. Employee A appears to have a higher score based on the average training score, but in fact, employee B has accumulated a total of 150 training points, while employee A has just 60.

In another situation, employee A only attended one training session and received a training score of 100 on average. Employee B completed three training sessions and received a training score of 25 on average. Employee A outnumbers employee B in terms of the number of training hours. Employee A has a total score of 100, whilst employee B only has a score of 75. A newly hired employee may require less training than a veteran of the organization. The two situations described above give birth to a third column, "total score", which provides an approximation of an employee's total score. This feature is the multiplication of two features: average training score and number of trainings. This column is useful for comparing and distinguishing between employees who have improved and those who have not.

The total score field is numeric and on the ratio scale. The goal is to see whether there are any correlations between the total score and the "is promoted" column. The total score column is separated into three bins (categories) for this purpose: Low (65 or below), Mediocre (65 to 145 points), and High (145 or higher). The bins were chosen based on the distribution of the total score column for the promoted workers. The column is divided into bins and placed in the total score label column for further analysis using the pd.cut () procedures.

Error! Reference source not found. resulted in the following conclusions:

- Employees who have been promoted have scores in the Mediocre and High ranges, i.e., 65 and above.

- Employees with low ratings have also been promoted a large percentage of the time.
- The fact that the lowest score level has the largest percentage of promoted employees indicates that the total score is not the only factor for promotion.

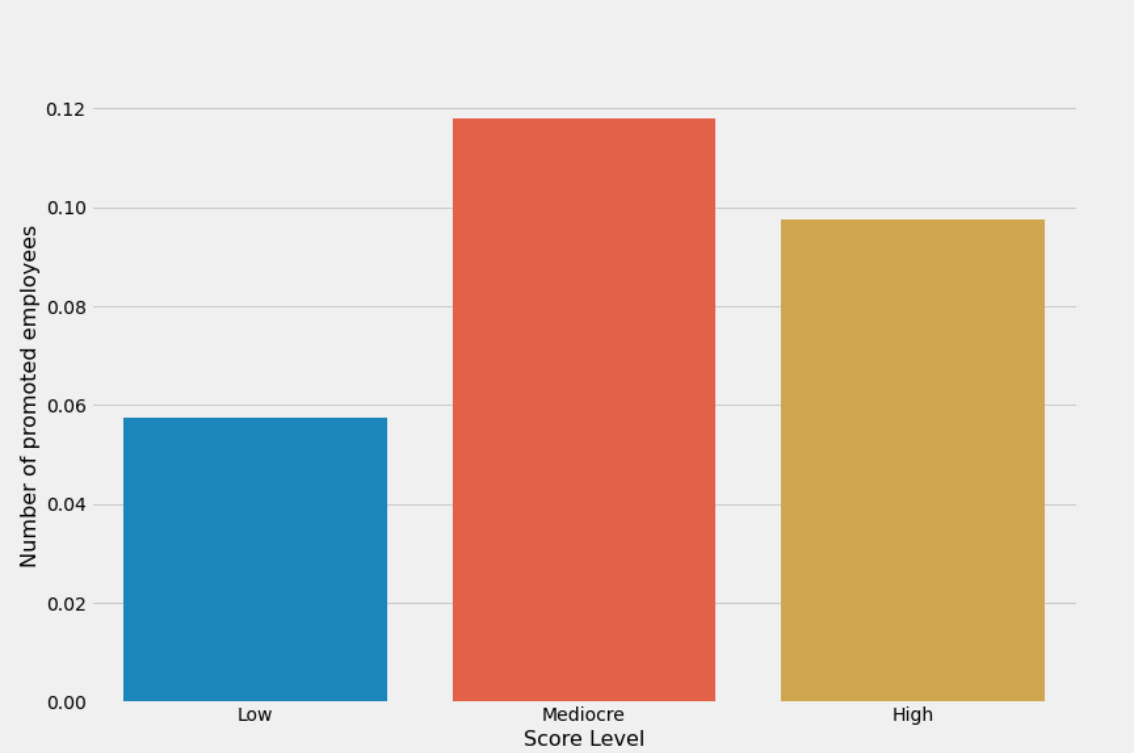


Figure 4.3 Score level distribution.

- **Work fraction:** this was a new feature that was created to represent the fraction of work done with their age.
- **Work start year:** this was another feature that represents the start age of the employee.

  Figure 4.4 shows that employees who start working at an earlier age between 24-29 have a higher chance of being promoted.

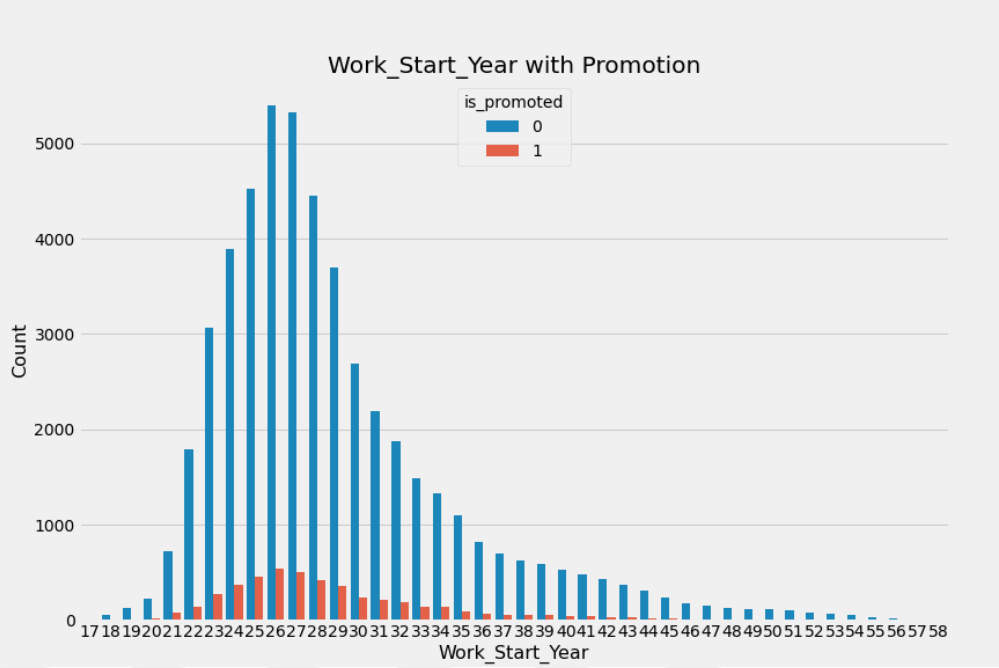


Figure 4.4 Work_Start_Year Distribution.

- **Years remaining to retire:** This is another new feature that will represent the remaining years for the employee until retirement.

  Figure 4.5 shows that employees having years remaining to retire between 25-33 have a higher chance to be promoted.

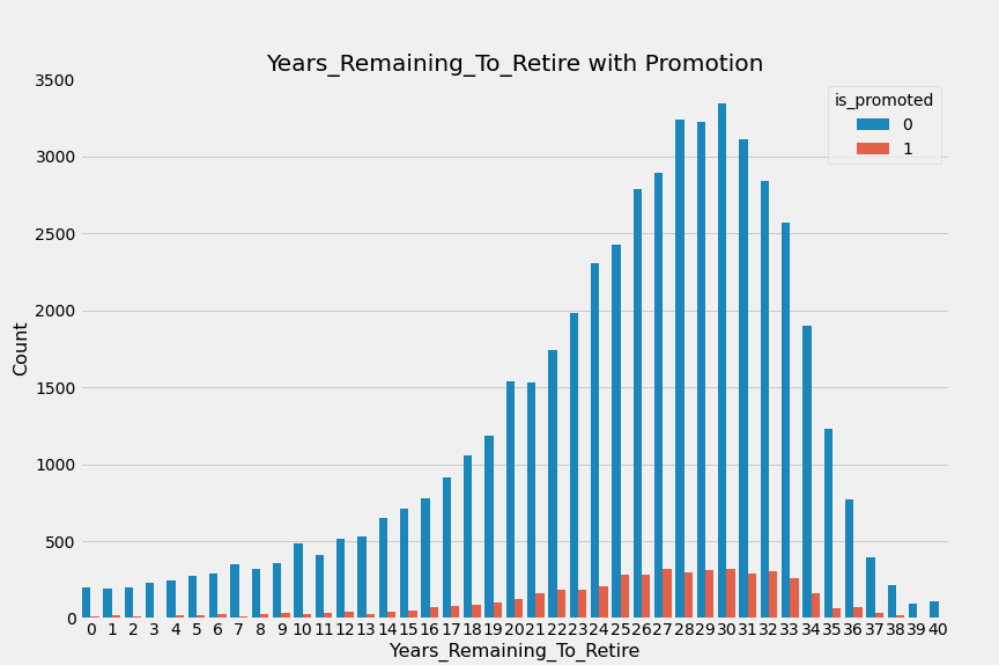


Figure 4.5 Years_remaining_to_retire distribution.

- **Performance:** For ease of analysis, the two columns KPIs_met and awards_won are combined into a single column performance using the any() function. Any employee who has either won an award or has met KPIs has shown good performance.

  Figure 4.6 compares counts of employees who were promoted against counts of employees who were not promoted. The following can be concluded:

- The majority of the individuals who were promoted had demonstrated excellent performance.
- Employees who were not promoted have a high rate of non-performance.

  Many employees who have worked hard yet have not been promoted. This might be related to a variety of different factors. This provides a solid reason to investigate the other aspects as well.

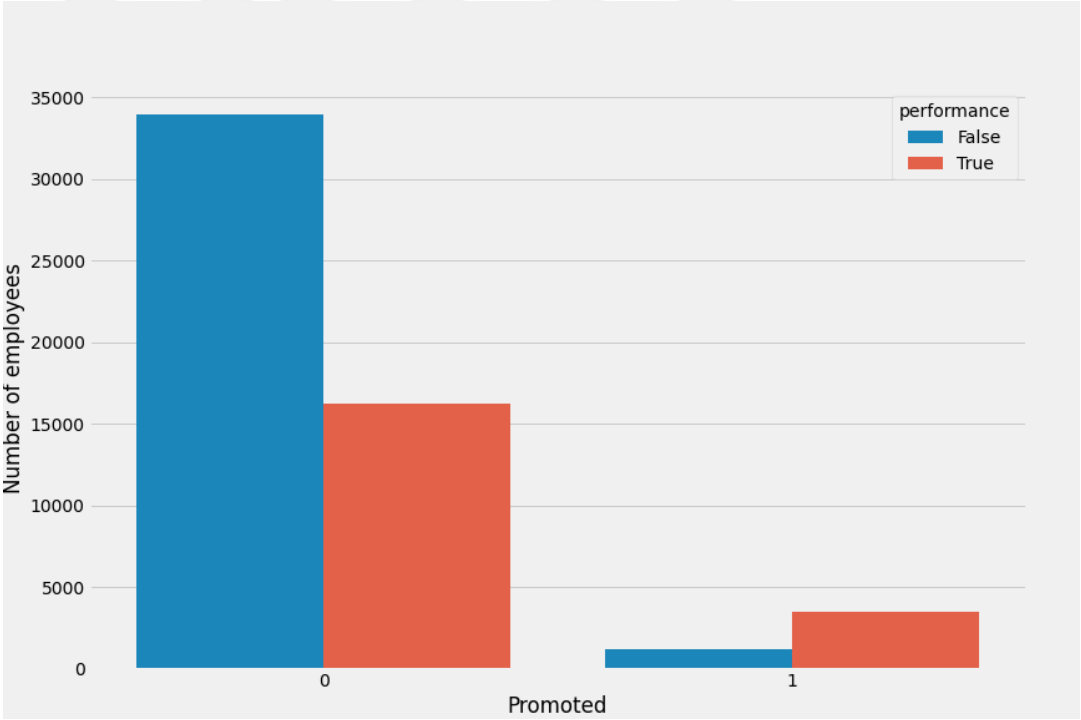


Figure 4.6 Performance distribution.

4. 2. 2 Binning the Numerical and Categorical Features

To have a good performance in this phase, we combine the levels of “no_of_trainings” which has fewer observations in train and test data. For the age feature, we bin ‘age’ data into groups (every 5 years as a bin) as shown in Figure 4.7.

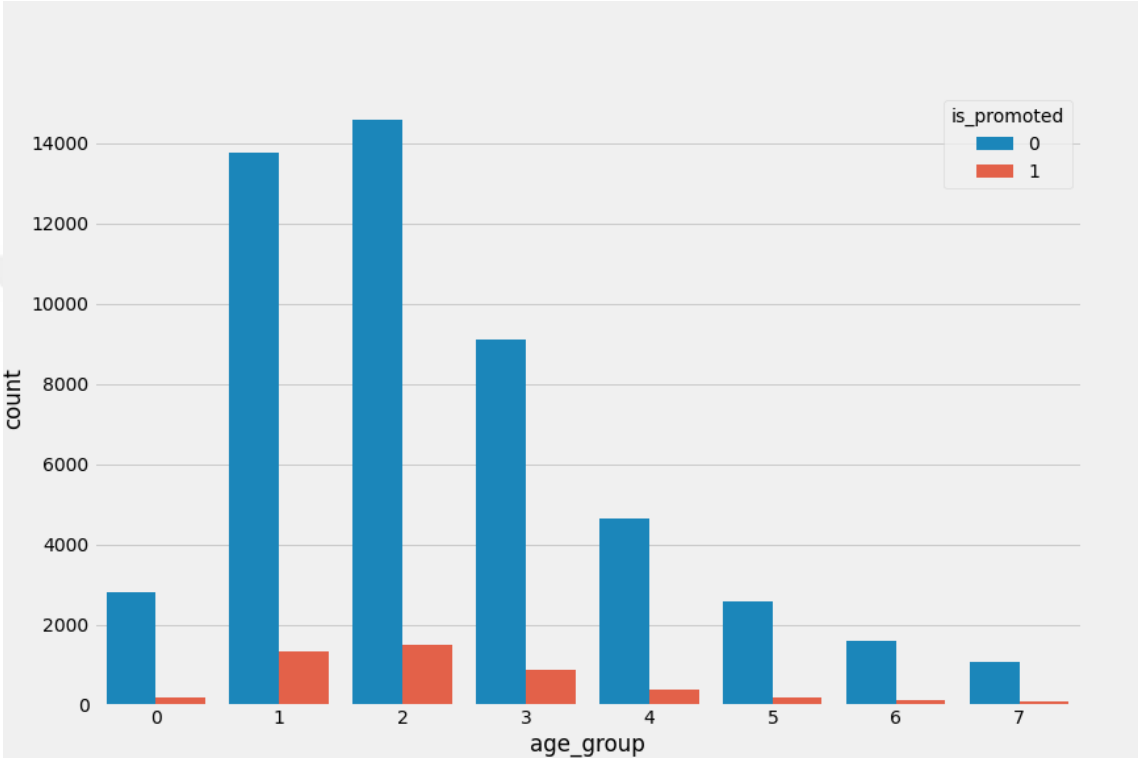


Figure 4.7 Age_group distribution.

4. 2. 3 Removing Unnecessary Feature

In this phase, we will delete some of the columns that are not relevant for forecasting promotion. Since we already know that the recruiting channel has very little to do with employee promotion, we will eliminate this column. Similarly, the region appears to have very little to do with promotion, so we will remove it as well. In addition, the column total_score_label was removed because this column was added only in order to better understand the total score feature.

We already know that machine learning algorithms only deal with numbers. As a result, we must encrypt our object data and convert it to numeric form in order for the machine

learning model to accept our data. Categorical variables are well-known for hiding and masking a wealth of important information in a data collection. It is critical to understand how to cope in such situations, otherwise, we would lose out on discovering the most essential variables in a model. Initially, we concentrated on numerical variables and then on categorical variables.

There are several methods for converting category columns to numerical columns. This is an important step because our machine learning models only function with numeric values. In this case, we will utilize Business Logic to encode the education column. The Label Encoder will then be used to encode the Department, Gender, and Number of Training columns. Thereafter, we can encode the categorical features into numerical form so that we can use them in our model.

Splitting the data is a critical stage in performing machine learning prediction on a dataset. We separated the Target and Independent Columns. By eliminating the target column from the data, we store the target variable in y and the remainder of the columns in x. In addition, for clarity, we are altering the name of data_test1 to x _test.

The train-test split is a method of assessing the performance of a machine learning system. It may be used for any supervised learning technique and can be utilized for classification or regression tasks. The goal is to assess the performance of the machine learning model on new data that was not used to train the model. The process entails splitting a dataset into two subgroups. The first subset, known as the training dataset, is utilized to fit the model. The second subset is not used to train the model; instead, the model is fed the input element of the dataset, and predictions are generated and compared to expected values. This second dataset is known as the test data.

- Train Dataset: This dataset is used to fit the machine learning model.
- Test Dataset: Used to assess the fit of the machine learning model.

This is how we want to put the model to use in practice. In other words, we want to fit it to existing data with known inputs and outputs and then make predictions on fresh cases in the future when we do not have the expected output or goal values. In order to conduct research, the dataset must contain a large number of characteristics that impact employee

advancement directly or indirectly. In actuality, we had to first select and trim the features into a dataset with a smaller number of attributes that were relevant to the study.

When creating a predictive model, feature selection is the process of minimizing the number of input variables. In general, it is preferable to limit the number of input variables to reduce modeling computational costs and, in certain situations, increase model performance. Statistical-based feature selection approaches include applying statistics to evaluate the relationship between each input variable and the target variable and selecting the input variables having the strongest link with the target variable. These strategies can be both quick and effective. Furthermore, the first and most critical phases in constructing our model should be feature selection and data cleansing. Irrelevant characteristics in our data can reduce model accuracy and cause our model to train based on irrelevant information.

Machine learning feature selection strategies may be roughly categorized into the following:

- Supervised Approaches: These techniques may be used on labeled data to discover significant features for improving the effectiveness of supervised models such as classification and regression. Wrapper, filter, and intrinsic supervised techniques are the three types of supervised methods.
- Unsupervised Techniques: These can be utilized with unlabeled data.

Filter techniques will be used in our model. Filter feature selection approaches employ statistical techniques to assess the connection between each input variable and the target variable, and the results are used to choose the input variables that will be employed in the model.

Using the model's feature importance attribute, we can determine the feature importance of each feature in our dataset. Characteristic significance assigns a score to each item of our data, with the higher the score indicating that the feature is more significant or relevant to our output variable. Feature importance is a built-in class in tree-based classifiers. We will use SelectKBest to extract the top features from the dataset. The SelectKBest technique chooses the features based on the highest score. We may use the

approach for both classification and regression data by altering the "score func" option. When preparing a big dataset for training, it is critical to select the optimal features. It allows us to discard less significant data and shorten the training time.

Table 4.2 gives the score of each item to help us determine which characteristics are the most significant. These rating aid in determining the optimal attributes to employ in our model.

Table 4.2 Scores of each feature.

	Feature	**Score**
0	department	0.001172
2	gender	0.301599
1	education	4.207178
11	work_fraction	5.744137
12	work_start_year	14.002753
15	age_group	17.419796
5	length_of_service	19.351084
13	years_remaining_to_retire	37.610788
4	previous_year_rating	574.656810
9	sum_metric	1538.912280
6	KPIs_met > 80%	1743.827117
7	qwards_won	2054.009313
14	performance	2155.495006
10	total_score	2851.455540
3	no_of_trainings	2972.721236
8	avg_training_score	5072.973743

We next wrap the model in a SelectFromModel instance using the feature importance derived from our training dataset. This is used to pick features on our training dataset, train a model using the XGBoost classifier using the selected subset of features, and then assess the model on the test set using the same feature selection strategy. We may test several thresholds for picking features based on feature relevance for interest. The feature importance of each input variable (**Figure 4.8**), in particular, allows us to rank each subset of features in order of relevance, starting with all features and ending with the most significant feature (Brownlee 2020). This will be further discussed in (5. 3 Experimental Results).

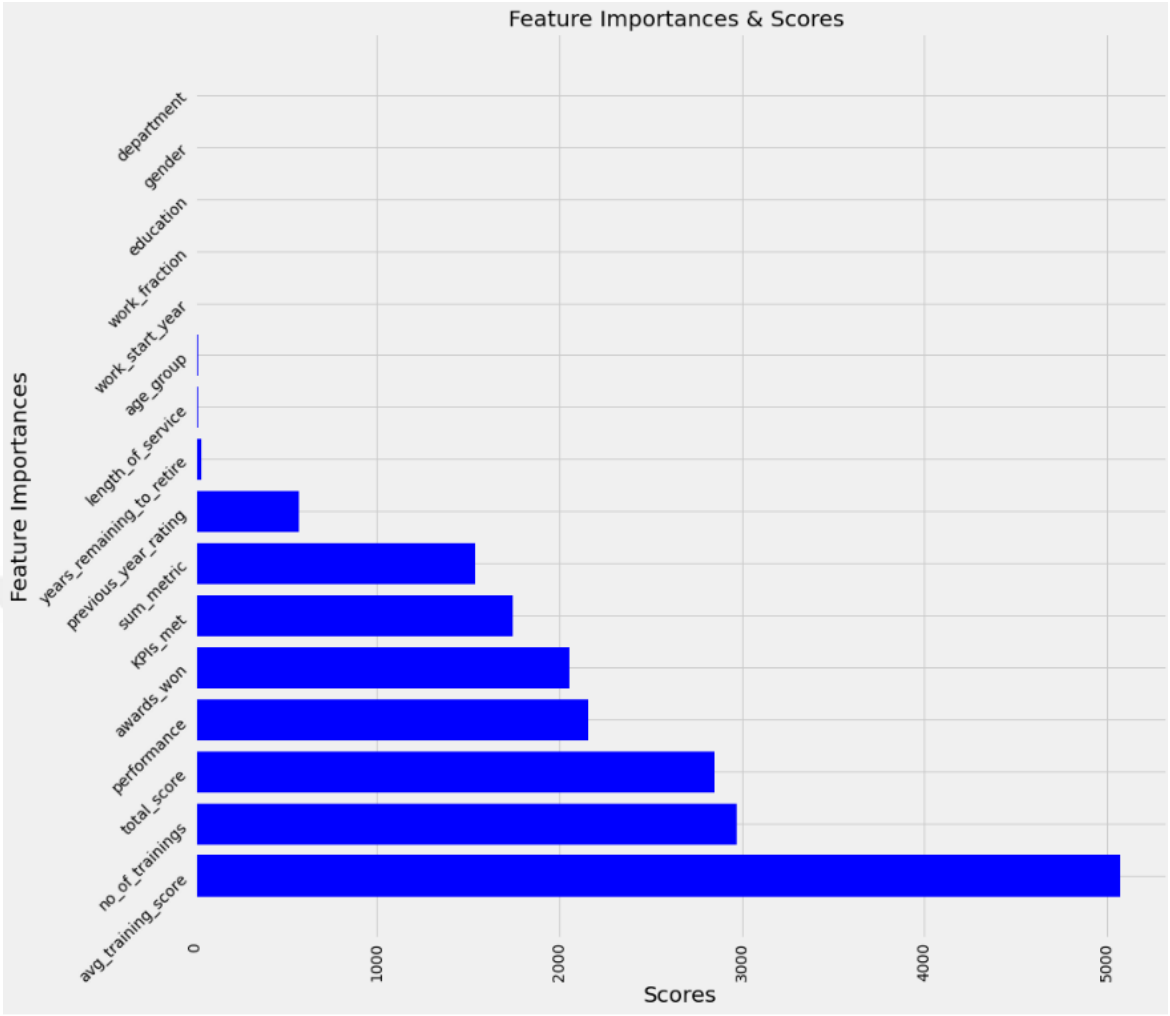


Figure 4.8 Feature importance and scores.

According to the results of this procedure shown in Table 4.3, ‘department’, ‘education’, and ‘gender’ are omitted. These features are necessarily the least important features for promotion prediction in our model. From the above conclusions, it can be stated with confidence that no factor alone is responsible for the promotion of an employee. The following factors together can be considered for predictive modeling:

Table 4.3 Factors considered for predictive modeling.

Factors considered for predictive modeling
no_of_trainings
previous_year_rating
length_of_service’
KPIs_met
awards_won
avg_training_score
sum_metric
total_score
work_fraction
work_start_year
years_remaining_to_retire
Performance
age_group

Furthermore, in the case where there is no parallel result of data processing for filtering, this means that the statistical analysis does not match the outcome of the ML algorithm. Therefore, another step is necessary to go back and check again. This step is to check the match of statistical analysis with the ML algorithm result in order to check the data if there is no good match between the two techniques.

4. 3 Data Manipulation: Preprocessing and Manipulate Data

A class imbalance develops when observation in one class exceeds observation in other courses. Class imbalance is a prominent issue in machine learning, particularly in classification issues. They are often divided into two classes: the majority (negative) class and the minority (positive) class. The majority of machine learning algorithms perform best when the number of samples in each class is about equal. This is since most algorithms are intended to enhance accuracy while minimizing mistakes.

Resampling is a technique that involves taking multiple samples from the original data samples. The resampling technique is a nonparametric statistical inference approach. There are several statistical methods available for resampling data, including oversampling, cluster-based sampling, and undersampling (Figure 4.9). In data analysis, oversampling and undersampling are approaches for adjusting the class distribution of a

data collection. These concepts are used in statistical sampling, survey design methods, and machine learning, among other places. Oversampling and undersampling are approaches that are opposed and nearly comparable. We saw earlier in this problem that the target column is highly unbalanced, so we need to balance the data using the over-sampling technique.

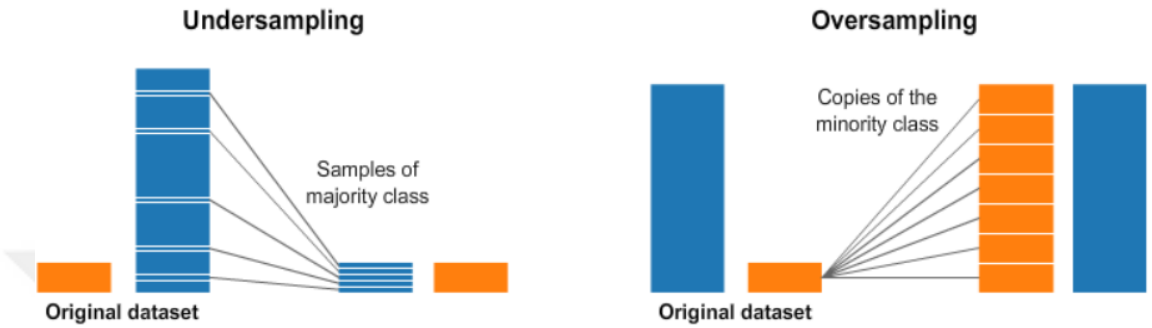


Figure 4.9 Undersampling & Oversampling technique.

SMOTE is regarded as one of the most prominent and important data sampling algorithms in ML and data mining. SMOTE oversamples the minority class by manufacturing "synthetic" cases rather than oversampling using replacement (Figure 4.10). These newly added synthetic examples are based on online segments connecting a defined number of k minority class nearest neighbors, which is set to five by default in the imblearn package.

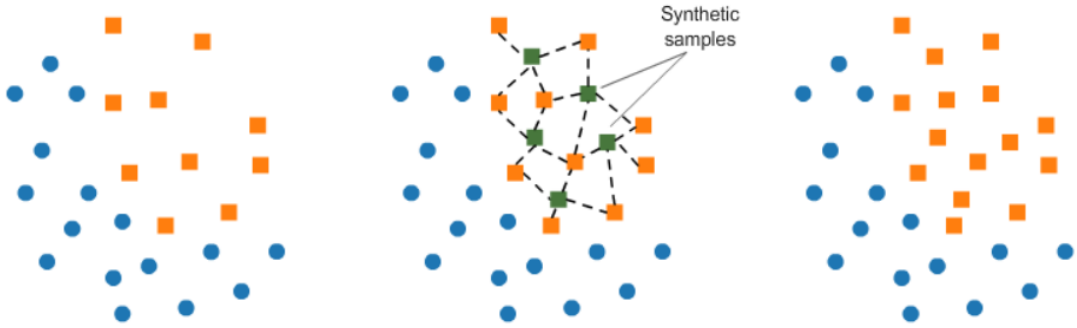


Figure 4.10 Synthetic Minority Oversampling Technique.

The SMOTE algorithm operates in four simple steps:

- As the input vector, select a minority group.
- Find its k closest neighbors (k neighbors is an input to the SMOTE() method).
- Select one of these neighbors and insert a synthetic point somewhere on the line connecting the point under consideration and its selected neighbor.
- Repeat the process until the data is balanced.

After balancing the data, we are now separating/splitting the entire dataset into training and testing data. The dataset is divided into two parts: training and testing, the former of which train the model, and the latter tests the model. Data will be used in training around 80% of the time, with the remaining 20% of data utilized to evaluate the model's performance.

Feature scaling is a strategy for lowering the values of all of the independent characteristics of the dataset on the same scale. Feature selection aids in the speed with which algorithms perform computations. Data processing is also known as data normalization and is performed during the data preparation step. It is a critical stage in our data preparation. If feature scaling is not performed, the machine learning model assigns more weight to higher values and less weight to lower ones. In addition, training the machine learning model takes a long time (SagarDhandare 2021).

Normalization, Robust Scalar, and Gaussian Transformation are all examples of feature scaling. The phrases "normalization" and "standardization" are used simultaneously at times, but they normally relate to separate phenomena.

a. Normalization

Normalization, Robust Scalar, and Gaussian Transformation are all examples of feature scaling. Normalization and standardization are used simultaneously at times, but they normally relate to separate phenomena. Normalization is a scaling technique in which values are rescaled from 0 to 1. There is also what is known as max/min normalization (min/max scaling). The smallest value of each characteristic is turned into 0, and the greatest value is transformed into 1 (SagarDhandare 2021), refer to Equation (4.1).

$$x_{new} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{4.1}$$

To normalize our data, we must import MinMaxScalar from the Sci-Kit learn library and apply it. After using the MinMaxScalar, the minimum and maximum values will be 0 and 1, respectively.

b. Standardization

Another scaling strategy is standardization, which has the mean equal to zero and the standard deviation equal to one. The characteristics will be rescaled as a consequence of standardization (or Z-score normalization) to guarantee that the mean and standard deviation are 0 and 1, respectively, refer to Equation (4.2). This approach to rescaling feature values with a distribution value between 0 and 1 is important for optimization algorithms like gradient descent, which are employed in machine-learning algorithms that weight inputs (e.g., regression and neural networks). Rescaling is also employed in algorithms that require distance measurements, such as the K-nearest-neighbors method (KNN) (SagarDhandare 2021).

$$X_{new} = \frac{X_i - X_{mean}}{\text{Standard Deviation}} \tag{4.2}$$

To standardize our data, we must import StandardScalar from the Sci-Kit learn library and apply it to it. However, whether to use normalization or standardization depends on the task and the machine learning method being used. There is no clear rule for determining when we should normalize or standardize our data. As a result, we begin by fitting our model to raw, adjusted, and standardized data and then compare performance to achieve the best outcomes. We, therefore, have decent performance using standardization rather than normalization. Furthermore, it is best to fit the scaler to the training data before using it to change the testing data. This prevents data leaks during the model testing procedure. In addition, scaling goal values is not always necessary.

4. 4 Data Modeling

Classification has two separate implications in machine learning. We may be provided a collection of observations to determine whether or not classes or clusters exist in the data, or we may be certain that there are a specific number of classes, and the goal is to devise rules that will allow us to categorize a new observation into one of the existing classes. The former is referred to as "unsupervised learning", whereas the latter is referred to as "supervised learning". Because the data is divided into two types—promoted and non-promoted—this work deals with classification as supervised learning. Every instance of the given problem would be classified as to whether the employee promotes the company or not.

The modeling process includes selecting models based on the different machine learning approaches utilized in the testing. In this scenario, multiple predictive models such as XGBoost, Random Forest, Decision Tree, Logistic Regression, AdaBoost, and Gradient Boosting were applied (ibrir, 2022). The objective is to find the best classifier for the problem under consideration. As a consequence, each classifier must be trained on the feature set, and the classifier with the best classification results is used to forecast. Section 3.2 discussed the categorization algorithms that were considered.

We continued with the creation of the prediction model to identify individuals who might potentially be promoted in the organization after outlining the objectives and appropriately preparing and analyzing the dataset to be utilized. In order to train the model to classify fresh observations, which would compose the test-set, a training set of examples from an already classified population (target) was required during the development phase of a model that implements a supervised learning algorithm. The model must next be trained on a constant amount of data to improve its prediction capabilities. The precision of machine learning algorithms grows in direct proportion to the amount of data provided during training. Ideally, there would be two different datasets: one for training and another for testing. Due to the lack of two specialized datasets in this situation, the original dataset was partitioned into two portions with 80% for training and 20 % for testing.

4. 5 Data Evaluation (fine & tune)

The above-mentioned data set includes features such as "performance", "no_of_training", "previous_year_rating", and so on. The learning algorithm will anticipate whether or not the employee will be promoted to the organization based on these values. The anticipated value is compared to the database's actual value. The "Confusion Matrix", which is a typical assessment criterion for any classification model, is utilized to evaluate the experimental outcomes. Using this, parameters such as accuracy, precision, recall, and F1-Measures are employed, and the appropriate values gained from experimentation are shown in the following section concerning various learning strategies.

The number of cases properly categorized by the model may be determined using a typical confusion matrix. The confusion matrix visualizes a classifier's performance, providing a complete analysis with data on the number of true positives, false positives, true negatives, and false negatives. If the dataset is uneven and untrustworthy, its only accuracy can produce deceptive findings. A classification report would show the model's accuracy, recall, Roc, AOC, and F1-Score. Precision and recall are based on the measure of relevance, with precision being the proportion of relevant samples found among the retrieved samples and recall indicating the fraction of relevant samples found among the total number of relevant samples. The classifiers are assessed using the mentioned evaluation criteria in order to choose the best model for the issue (Jain, Jain, and Pamula, 2020).

When the datasets are divided, it is critical to maintaining the same distribution of target variables across both the training and test datasets. As a result, it is critical to prevent having a random subdivision change the proportion of classes present in the training and test datasets from the original. The goal *is promoted* attribute is a binary variable with 91% "No" and 9% "Yes", with both datasets retaining the same percentage after splitting.

In order to find the optimal model for the issue, the classifiers are evaluated using the evaluation criteria listed above. The model will then be fine-tuned repeatedly using Hyperparameter Tuning and Cross-Validation until we discover the optimal model.

A good cross-validation (CV) approach should be used (Figure 4.11), which is the most crucial part of modeling. This research utilized a five-fold CV. Cross-validation is a strategy for preventing over-fitting and simplifying the model. The training set was randomly divided into five parts (k), with one serving as a validation set and the other k-1s serving as training sets, and the operation was repeated k times. Each iteration used a different section as the validation set, and the average prediction error was calculated by averaging the average errors in the k-validation sets (Fallucchi et al. 2020).

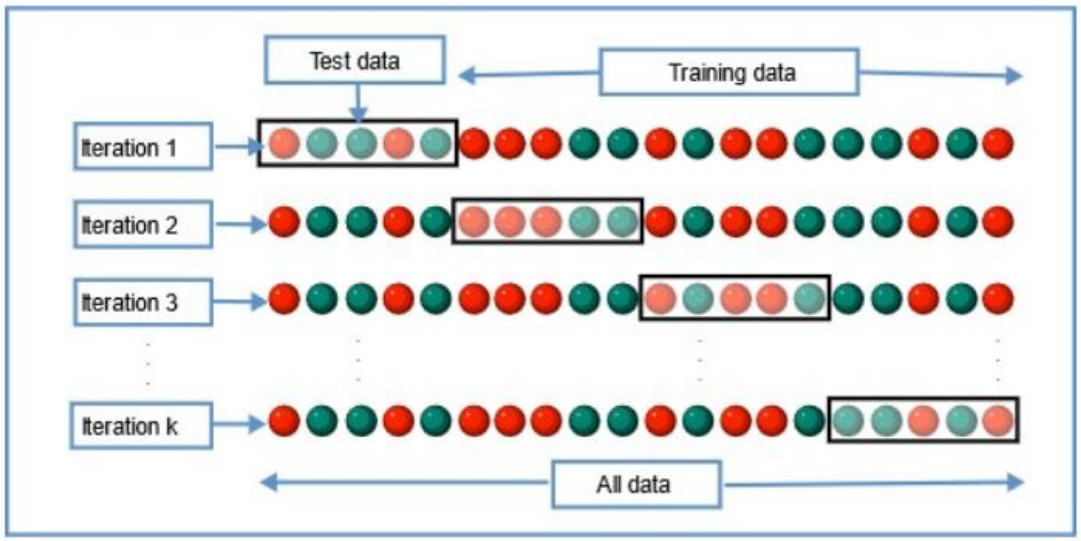


Figure 4.11 Cross-validation strategy.

The complete dataset will be cross-validated on the full dataset using the sklearn.model selection and cross-validation method to check how the model performs on this dataset. Cross-validation is an effective strategy for model selection. The model that performs well in cross-validation is then selected for additional training, testing, and hyper-parameter adjustment.

The next stage is to fine-tune the hyper-parameters. This optimizes the model's parameters to the optimum set that maximizes the model's accuracy. The ML model used in this case is the XGBoost Classifier. XGBoost is a powerful machine learning method, particularly in terms of speed and accuracy. To increase and completely use the XGBoost model's benefits over competing algorithms, parameter adjustment is required. A grid search must be performed for all relevant model parameters. There are numerous settings,

especially in the case of XGBoost, and it may be fairly CPU-expensive at times (Aarshay 2020).

XGBoost settings have been classified into three groups by the creators of XGBoost (Aarshay 2020):

- General parameters: Direct the overall operation.
- Booster parameters: At each stage, direct the specific booster (tree/regression).
- Learning Task Parameters: Direct the optimization process.

a) General parameters

Table 4.4 determines XGBoost's general functionality.

Table 4.4 General parameters of XGBoost.

[default=gbtree] booster	➢ Choose the model to run at each iteration. There are two options: ▪ gbtree: models based on trees ▪ linear models, gblinear
[default=0] silent:	➢ When silent mode is enabled, the value is set to 1, which means that no running messages are produced. ➢ It is typically a good idea to maintain it at zero because the messages may aid in understanding the model.
nthread [defaults to the maximum number of available threads if not specified]	➢ This is utilized for parallel processing, and the number of system cores should be specified. ➢ In case running all cores, no value should be supplied, and the algorithm will identify it for us.

b) Booster parameters

Though there are two types of boosters (Table 4.5), only the tree booster is considered here because it always outperforms the linear booster and so the latter is rarely employed.

Table 4.5 Booster parameters of XGBoost.

learning_rate **[**default=0.3]	➢ In GBM, this is analogous to the learning rate. ➢ Increases the model's robustness by decreasing the weights at each step. ➢ The following are typical final values to be used: 0.01-0.2
min_child_weight [default=1]	➢ Defines the minimal weighted total of all needed observations in a kid. ➢ This is comparable to but not the same as min child leaf in GBM. This relates to the minimum "total of weights" of the data, whereas GBM has a minimum "number of observations." ➢ Used to prevent over-fitting. Higher values prohibit a model from learning relations that are particularly unique to the sample used for a tree. ➢ Large numbers can result in under-fitting, so this should be controlled using CV.
max_depth [default=6]	➢ The maximum depth of a tree is the same as GBM. ➢ Higher depth allows the model to learn highly specific relations to a single sample, which helps to control over-fitting. ➢ Should be fine-tuned using CV. ➢ 3-10 are typical values.
max_leaf_nodes	➢ A tree's maximum number of terminal nodes or leaves. ➢ In place of max depth, this variable can be specified. Because binary

	trees are generated, a depth of 'n' would result in a maximum of 2n leaves. ➢ If this is set, GBM will disregard max depth.
gamma [default=0]	➢ Only when the ensuing split results in a positive decrease in the loss function is a node split. Gamma determines the smallest loss reduction necessary to split. ➢ This makes the algorithm more cautious. The values can and should change based on the loss function.
max_delta_step [default=0]	➢ We enable each tree's weight estimation to be in the greatest delta step. If the value is set to 0, this indicates that there is no limitation. It can assist make the update step more cautious if it is set to a positive number. ➢ Normally, this option is not required, however, it may be useful in logistic regression if the class is very unbalanced.
subsample [default=1]	➢ The same as the GBM subsample. Denotes the proportion of data that will be sampled at random for each tree. ➢ Lower values make the algorithm more conservative and avoid overfitting, while too low values may result in underfitting. ➢ Typical ranges: 0.5 to 1.
colsample_bytree [default=1]	➢ In GBM, this is equivalent to max features. Denotes the proportion of columns that will be sampled at random for each tree. ➢ Typical ranges: 0.5 to 1.
colsample_bylevel [default=1]	➢ Denotes the column subsample ratio for each split in each level.
Reg_lambda [default=1]	➢ On weights, there is an L2 regularization term (analogous to Ridge regression)

	➢ This was responsible for XGBoost's regularization. Though many data scientists do not utilize it frequently, it should be investigated to avoid overfitting.
reg_alpha [default=0]	➢ Weight L1 regularization term (analogous to Lasso regression) ➢ In the case of very high dimensionality, this can be utilized to make the method run quicker when implemented.
scale_pos_weight [default=1]	➢ When there is a large class imbalance, a number greater than 0 should be utilized to aid in speedier convergence.

c) Learning Task Parameters

These parameters determine the optimization aim and the measure that will be calculated at each stage shown in Table 4.6.

Table 4.6 Learning Task Parameters of XGBoost.

objective [default=reg:linear]	➢ This specifies the loss function that must be minimized. The most often used values are: ▪ binary: logistic – For binary classification, logistic regression yields predicted probability (not class) ▪ multi: softmax –multiclass classification with the softmax goal yields projected class (not probabilities) ▪ In the num class (number of classes) option, the number of distinct classes must be provided. ▪ multi:softprob –same as softmax, but gives the projected probability of each data point in each class.

eval_metric [default according to objective]	➢ The measure to be used for data validation. ➢ The default settings for regression and classification are rmse and error, respectively. ➢ The following are typical values: ▪ **rmse** is an abbreviation for root mean square error. ▪ **mae** is an abbreviation for mean absolute error. ▪ **logloss** — negative log-likelihood error – the rate of binary classification mistake (0.5 thresholds) ▪ **merror** — Error rate in multiclass classification ▪ **mlogloss** stands for Multiclass Logloss. ▪ **AUC** is an abbreviation for Area Under the Curve.
seed [default=0]	➢ The seed is a random number. ➢ It may be used to generate reproducible findings as well as to tune parameters.

The next step is using the general approach for parameter tuning. The various steps to be performed are:

- Selecting a somewhat fast learning rate. In general, a learning rate of 0.1 is adequate, although values ranging from 0.05 to 0.3 should suffice for various problems. Determine the best number of trees to use for this learning rate. XGBoost has a very handy function called "cv" that does cross-validation at each boosting iteration and hence delivers the optimum number of trees needed. In our study, the learning_rate was fixed to 0.1 and cv to 5.
- Tree-specific parameters (max depth, min child weight, gamma, subsample, colsample by tree) should be fine-tuned for the chosen learning rate and the number of trees. Here, after many iterations and tuning with changing in different values and looking at the performance, we fixed finally these values: max_depth=4, min_child_weight=6, gamma=0.1, subsample=0.8, colsample_bytree=0.8.

- Regularization settings (lambda, reg_alpha=0.01) for XGBoost can be adjusted to minimize model complexity and improve performance.
- 'scale pos weight' is one of the most critical factors that people frequently overlook when dealing with an unbalanced dataset. This parameter should be fine-tuned with caution, since it may result in overfitting the data.

5. RESULTS AND DISCUSSIONS

Details of the experimental settings, assessment measures, and classification experiment outcomes are described in this chapter.

5. 1 Experimental Setup and Details for Experiments

Supervised learning is used to predict employee promotion. Models were built using XGBoost, Random Forest, Decision Tree, Logistic Regression, AdaBoost, and Gradient Boosting. They are effective in a variety of settings and have addressed several prediction issues. The promotion prediction experiment was conducted in Python. We begin by extracting characteristics. Furthermore, use a feature selection technique to discover the most significant features. The feature set includes all features related to employee promotion. To divide the training and test sets, five-fold cross-validation is employed. To address the problem of class imbalance, the synthetic minority over-sampling method (SMOTE) is used. Grid search is used to fine-tune hyperparameters to find the best classifier. Model performance is measured using accuracy, the area under the curve, recall, F1 score, and precision.

5. 2 Evaluation Metrics

The collected test results of our research are assessed and contrasted using statistical criteria like accuracy, precision, recall, and F1 score. The terms "True Positive" (TP), "True Negative" (TN), "False Positive" (FP), and "False Negative" (FN) used in binary classification are given as a confusion matrix in Table 5.1. In our scenario, a positive instance corresponds to an instance with the class label "1" (promoted), while a negative instance relates to an instance with the class label "0" (unpromoted).

Table 5.1 Confusion matrix description.

	Positive Instance	**Negative Instance**
Classified as Positive	True Positive	False Positive
Classified as Negative	False Negative	True Negative

Accuracy is simply the percentage of correctly categorized occurrences (employee profiles) divided by the total number of instances. It refers to a measure's proximity to real value. Equation (5.1) may be used to compute it.

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{|\text{Positive Instance}| + |\text{Negative Instance}|} \quad (5.1)$$

Precision is the likelihood that a (randomly chosen) positively classified event is, in fact, positive. A precision rating of 1.0 indicates that every instance categorized as positive is, in fact, a positive case, but it does not indicate whether all positive examples are retrieved. Equation 5.2) contains the precision equation.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \quad 5.2)$$

The likelihood that a (randomly chosen) actual positive case is properly identified is referred to as "recall." A perfect recall rating of 1.0 indicates that all positive events are categorized as positive, but it does not specify how many negative examples are likewise labeled as positive (False Positive). Equation (5.3) may be used to calculate the recall value.

$$Recall = \frac{True\ Positive}{|Positive\ Instance|} \quad (5.3)$$

The harmonic mean of accuracy and recall is used to get the F1 score (balanced F-score). As indicated in Equation (5.4), the F1 score may be determined using precision and recall.

$$F_1\ score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5.4)$$

The Area under the receiver operating characteristic curve (ROC-AUC) is also used to compare classification accuracy. The AUC is a broad measure of 'predictiveness' that

decouples classifier evaluation from operational parameters such as class distributions and misclassification costs. Furthermore, AUC is better than other metrics such as error rate since it indicates the likelihood that a classifier ranks a randomly chosen positive instance higher than a randomly picked negative one.

Even when the experiment outcomes are compared using all of these measures, accuracy and F1 score are the ones that we focus on the most.

5. 3 Experimental Results

This phase assessed the suitability of the models used. But first and foremost, we had to select the appropriate variables for our work, so as we mentioned earlier in 4. 2. 3 Removing Unnecessary Feature) about the importance of selecting the feature, where we will display the results and emphasize the use of the XGBoost classifier in selecting those features because we deemed it more important. This is performed by using the SelectFromModel class, which receives a model and may convert a dataset into a subset with defined attributes. This class can take a pre-trained model, such as one that was trained on the whole training dataset. It may then choose which features to select by applying a threshold. This threshold is used when you use the transform() method on the SelectFromModel instance to consistently choose the same features on the training and test datasets (Jason Brownlee, 2016). In our scenario, we first train and then test an XGBoost model on the whole training and test datasets.

Table 5.2 demonstrates that the model's performance typically improves as the number of selected characteristics increases, starting with feature number seven, which has an accuracy of 83.68%. For this problem, there is a trade-off between features and test set accuracy, and we could decide to take a complex model (larger attributes such as n = 13) and accept a modest decrease in estimated accuracy from 84.21% to 83.96%, which is likely to be more useful based on the importance of the variables used, and, of course, the accuracy will improve more using grid search as the model evaluation scheme.

Table 5.2 Evaluation procedure with a number of selected features.

Threshold	Features Number	Accuracy
0.414	n =1	77.28%
0.150	n =2	77.28%
0.073	n =3	77.63%
0.073	n =4	78.66%
0.070	n =5	80.11%
0.040	n=6	80.08%
0.037	n=7	83.68%
0.026	n=8	84.04%
0.024	n=9	83.87%
0.018	n=10	83.92%
0.018	n=11	84.10%
0.014	n=12	83.93%
0.013	n=13	83.96%
0.012	n=14	84.16%
0.012	n =15	84.19%
0.006	n =16	84.21%

Following that, after selecting the 13 features to be used in the model, but this is insufficient for us, the next phase is to run the model and compare the results without the features that correlate, such as length of service and age being highly associated, as we discovered earlier in 3. 2. 3 Multivariate analysis). It can also be noticed that KPIs and previous year's ratings are correlated to some extent, signaling that there is some linkage, thus we eliminate those two characteristics of age and KPIs to avoid multicollinearity. As a result, the following ten characteristics will be used in running the six models: 'the number of trainings', 'previous year rating', 'length of service', 'awards won', 'avg training score', 'sum metric', 'total score', 'work fraction', 'work start year', and 'years remaining.

Comparing the results from Table 5.3 and Table 5.4, clearly seen that having high accuracy with the 13 features rather than 11 features. Therefore, we decided to choose 13 features to be employed in the model, Hence, the outcomes of the prediction phase judgments were gathered in the relative "confusion matrix," initially without using a grid search and then using a grid search for each method. This is a matrix in which the classifier's predicted values are given in the columns and the actual values of each instance of the test-set are shown in the rows. To start with the performance evaluation, we used the confusion matrix to generate a set of essential metrics to quantify the

efficiency of each algorithm: accuracy, precision, recall, and F1 score. Table 5.4 summarizes these measures, which are based on the number of mistakes and accurate responses generated by the classifier.

Table 5.3 Evaluation metrics with 11 Features of different classifiers.

Classifiers	Accuracy
Logistic Regression	0.85%
Decision Tree	0.89%
Random Forest	0.91%
AdaBoost	0.72%
Gradient Boosting	0.76%
XGBoost	0.75%

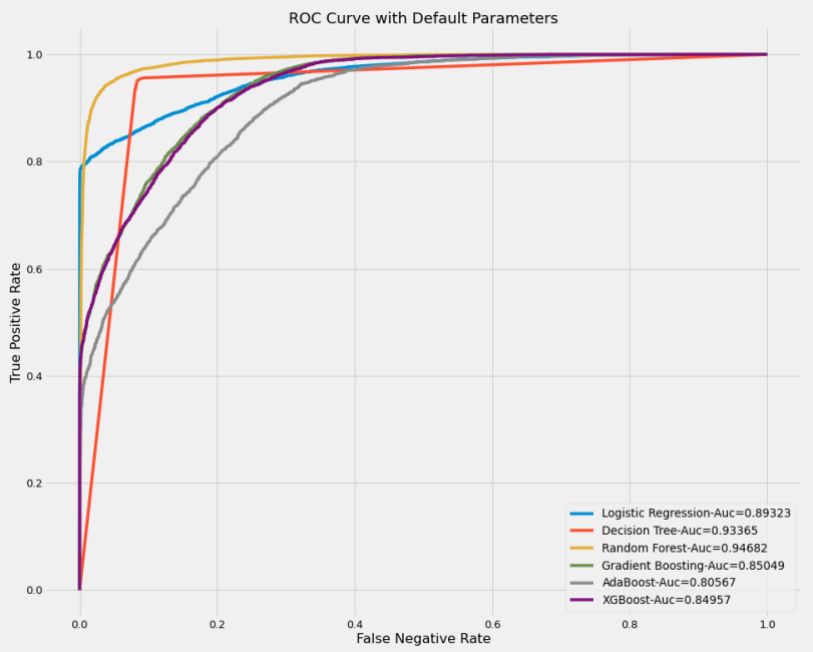
Table 5.4 Evaluation metrics with default parameters of different classifiers.

Classifiers	Accuracy	Precision	Recall	F1-score	ROC AUC
Logistic Regression	0.88%	0.88%	0.88%	0.87%	0.87%
Decision Tree	0.92%	0.91%	0.91%	0.91%	0.91%
Random Forest	0.93%	0.92%	0.92%	0.92%	0.92%
AdaBoost	0.80%	0.79%	0.79%	0.79%	0.79%
Gradient Boosting	0.82%	0.82%	0.82%	0.82%	0.82%
XGBoost	0.82%	0.77%	0.77%	0.77%	0.77%

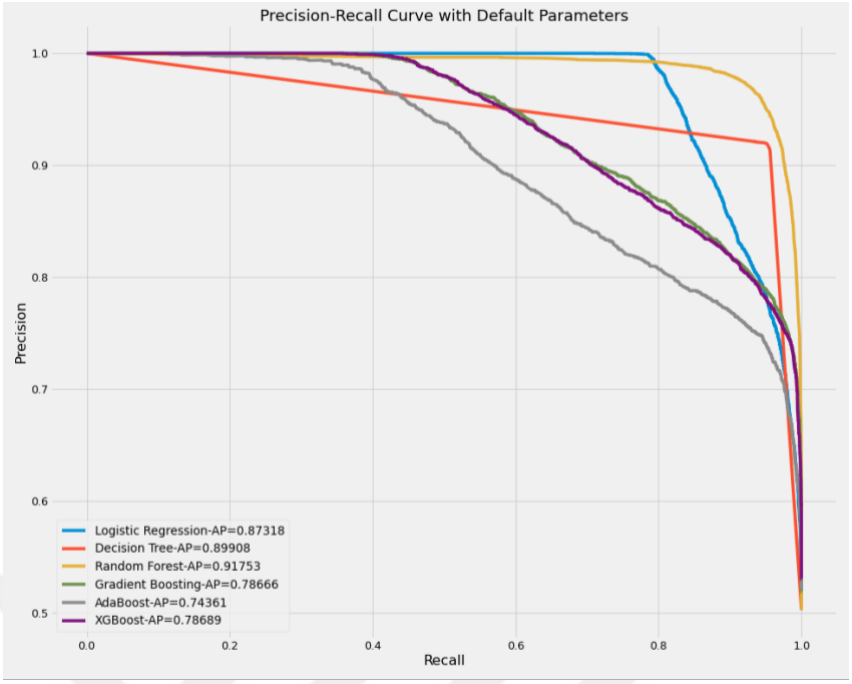
The results of this experiment revealed that all the classifiers had acceptable accuracy, which is greater than 80%. In many situations, this level of accuracy is seen as adequate. The dataset yielded acceptable models for each of the specified classification techniques in this experiment. The accuracy of the model is used to select the most acceptable classifier for the dataset to choose the appropriate classifier. As demonstrated in Table 5.4 the Random Forest classifier has the best accuracy, with 93%, the highest among the chosen classifiers. Similarly, when compared to other classifiers, Random Forest scores well on other metrics such as precision or recall, F-Measure, and ROC AUC. However, the AdaBoost Classifier model is less accurate than others, with an accuracy of just over 80%. With 92% accuracy, the decision tree also performed well. Figure 5.1shows the same conclusion in terms of the ROC AUC graph. According to Figure 5.1, the random forest has the highest average precision, i.e. true positive rate.

Figure 5.1 depicts the receiver operating characteristic (ROC) curve for several classifiers using false positive rate (FPR) and true positive rate (TPR). The bigger the area under the curve, the better the classifiers' accuracy. When the classifiers are evaluated using the confusion matrix, it is discovered that the RF achieves even greater accuracy than the DT, surpassing all of the other classifiers. One reason RF outperforms DT is that DT utilizes the whole sample in each step, selecting decision boundaries at random rather than selecting the best one. From the data, it is clear that RF has an accuracy rate of 93%. The accuracy of DT and RF is substantially higher, and it appears that these classifiers may be used to forecast whether an individual will be promoted within the organization. However, in our research, we will focus more on the XGBoost classifier because it is the uniqueness of our study and the first time using this classifier to forecast such a problem; thus, we will use grid search to improve the accuracy of XGBoost beyond 82%.

Considering model hyper-parameters impact performance, we alter the parameters of five models using grid search and cross-validation, with a heavy emphasis on the XGBoost classifier. The basic concept behind this approach is to select numerous parameter combinations in advance and run cross-validation for each set of parameters to discover the ideal parameter combination for XGBoost using five-fold cross-validation. This part is one of the strongest parts of this study concerning previous studies.



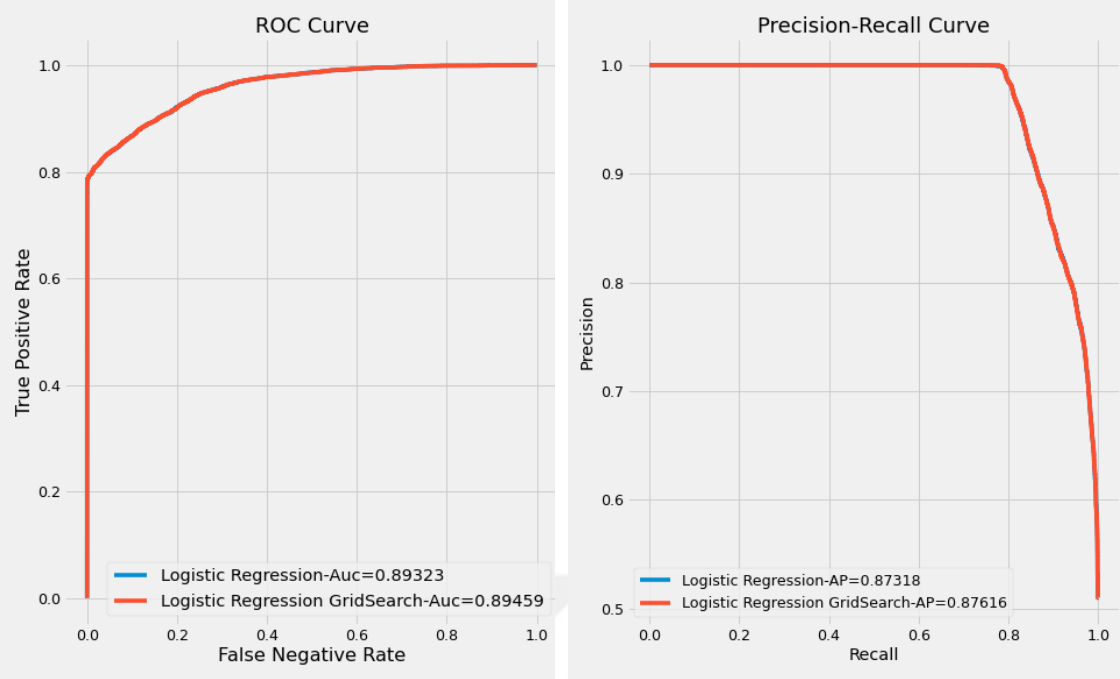
a)



b)

Figure 5.1 Default parameters of different classifiers for a) ROC b) Precision-Recall Curve.

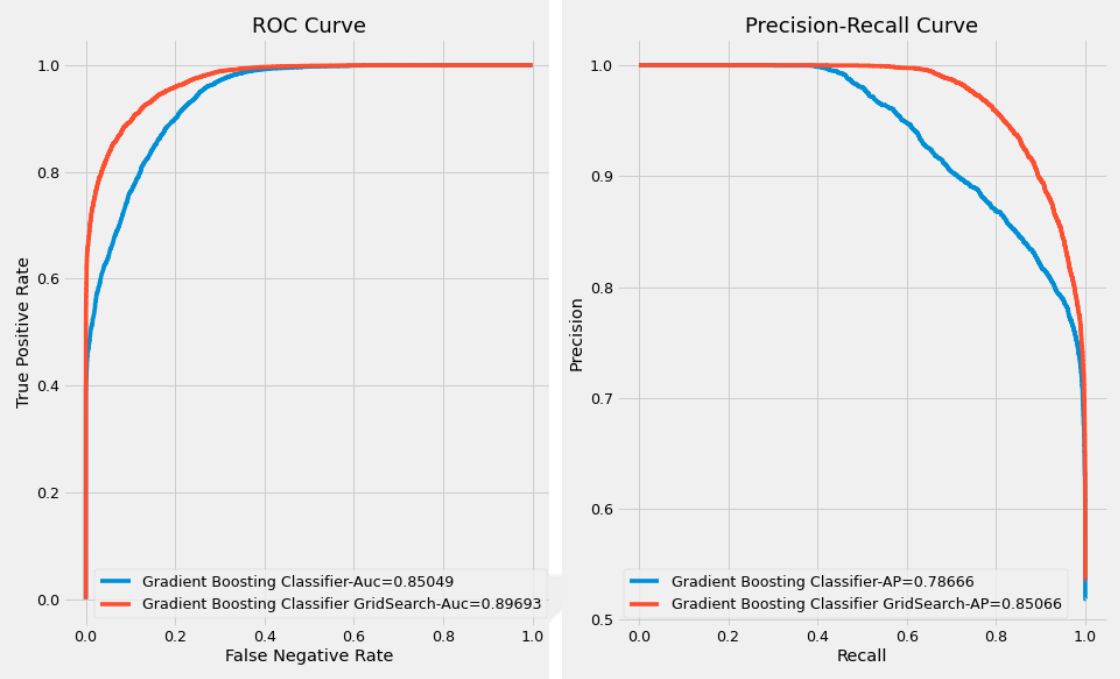
Figure 5.2 illustrates the ROC & Precision-Recall Curve with and without Grid Search of the Logistic Regression classifier. Both versions of the classifier seem to do a pretty good job, with an accuracy of 88%, but the logistic regression with the grid search version appears to perform slightly better.



a) b)

Figure 5.2 Result of Logistic Regression with & without Grid Search for a) ROC curve b) Precision-Recall Curve.

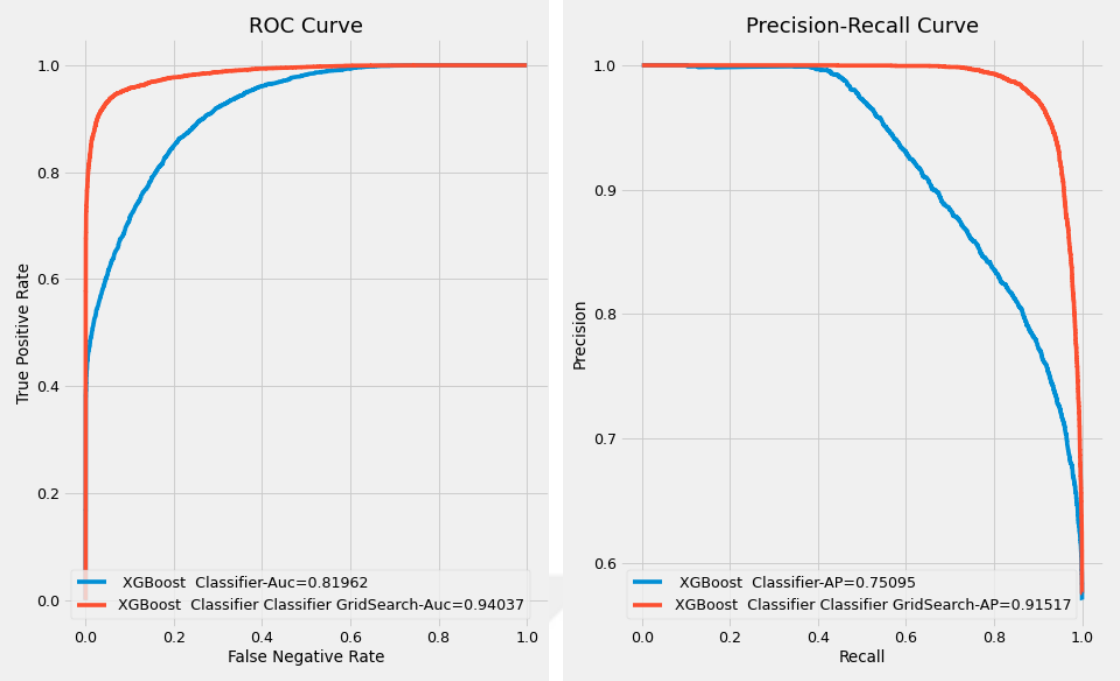
Figure 5.3 shows the ROC and Precision-Recall Curve with and without Grid Search of the Gradient Boosting classifier. It is seen that the accuracy of the grid search has improved, and it is better than the baseline model, with an accuracy of 85.81%.



a) b)

Figure 5.3 Result of Gradient Boosting with & without Grid Search for a) ROC b) Precision-Recall Curve.

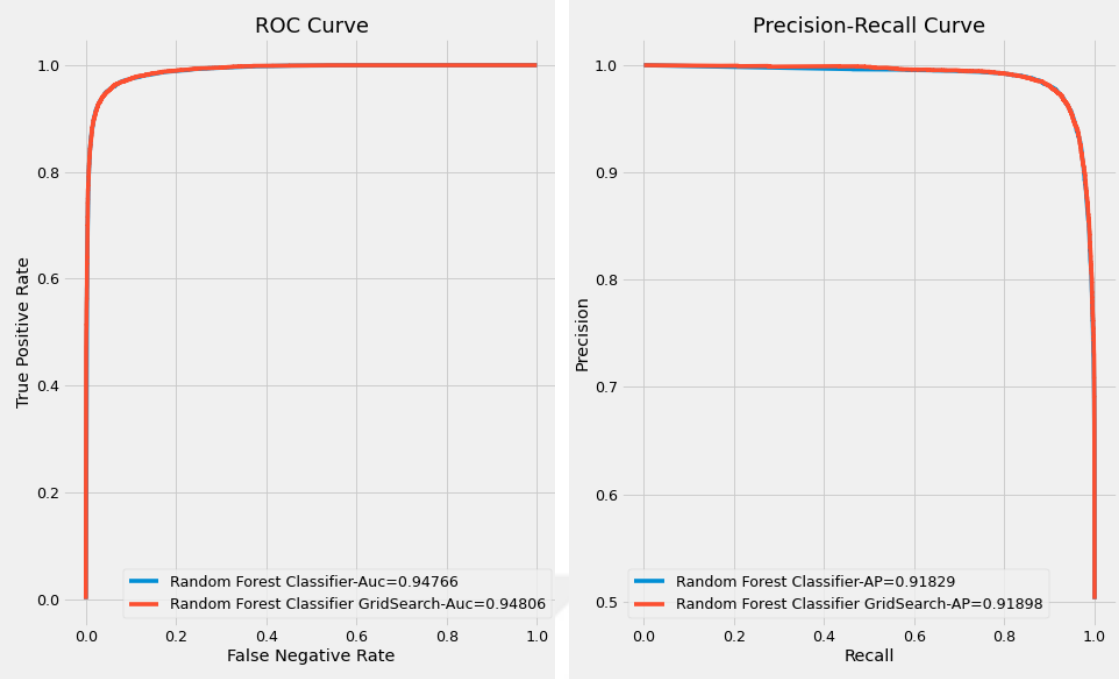
Figure 5.4 shows the ROC and precision-recall curve with and without grid search of the XGBoost classifier. As expected, the accuracy improved a lot using the grid search, which seems to do a pretty good job with 94.03% accuracy. Similarly, the precision-recall curve performs better, with 91.51%, with grid search.



a) b)

Figure 5.4 Result of XGBoost with & without Grid Search for a) ROC b) Precision-Recall Curve.

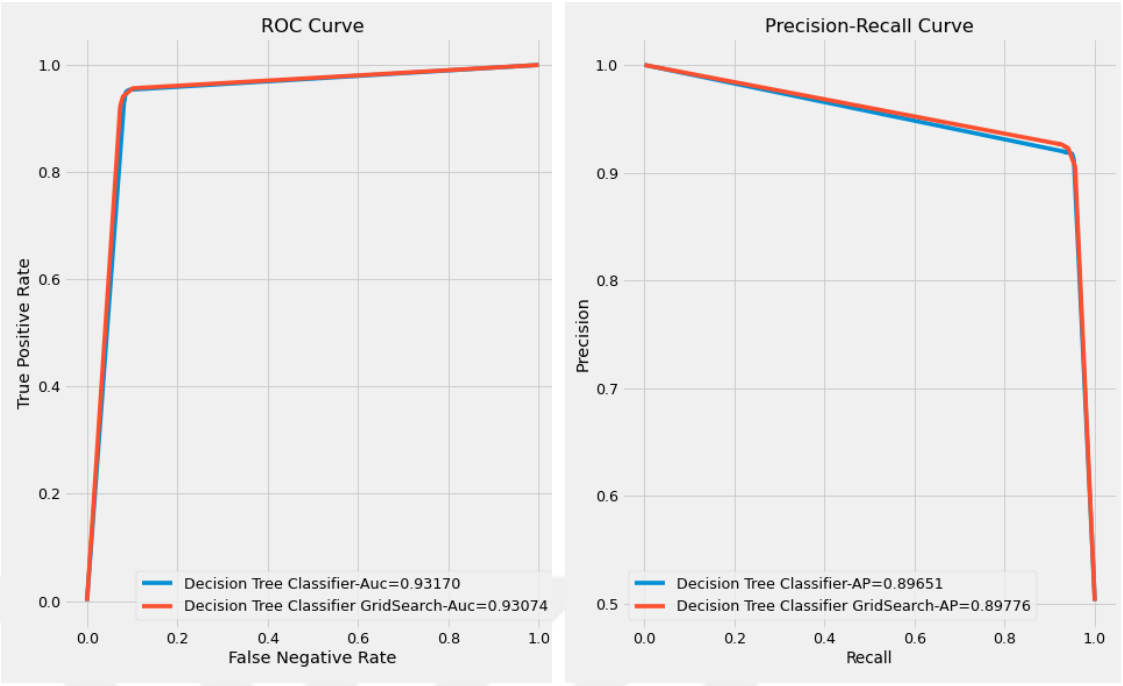
Figure 5.5 displays the ROC and Precision-Recall Curve with and without Grid Search of the Random Forest classifier. Both versions of the classifier seem to do a pretty good job, with an accuracy of 93%. This concludes that the performance did not improve using the grid search method. Also, the classifiers both have similar precision-recall curve scores of 89%.



a) b)

Figure 5.5 Result of Random Forest with & without Grid Search for a) ROC b) Precision-Recall Curve.

Figure 5.6 **Error! Reference source not found.** depicts the ROC and precision-recall curves with and without grid search of the decision tree classifier. Both versions of the classifier seem to do a pretty good job, with an accuracy of 91%. This concludes that the performance did not improve using the grid search method. Also, the classifiers both have similar precision-recall curve scores of 88%.



a) b)

Figure 5.6 Result of Decision Tree with & without Grid Search for a) ROC b) Precision-Recall Curve.

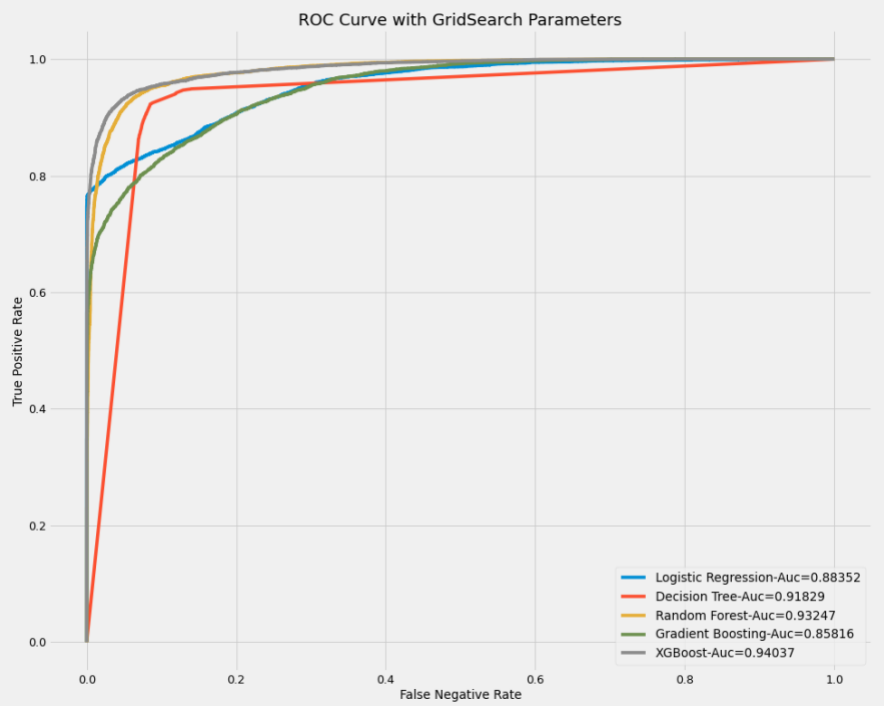
The ROC curve determines the accuracy of predicting data points in a class. The farther the graph is from the 45-degree line, the more accurate the forecast. As a result, the greater the Area Under the Curve (from 0 to 1), the better the outcomes.

Figure 5.2 - Figure 5.6 show a comparison of the ROC and Precision-Recall Curve with and without grid search for each of the five classifiers listed above. According to the ROC curve, the XGBoost model outperforms the other four classifiers.

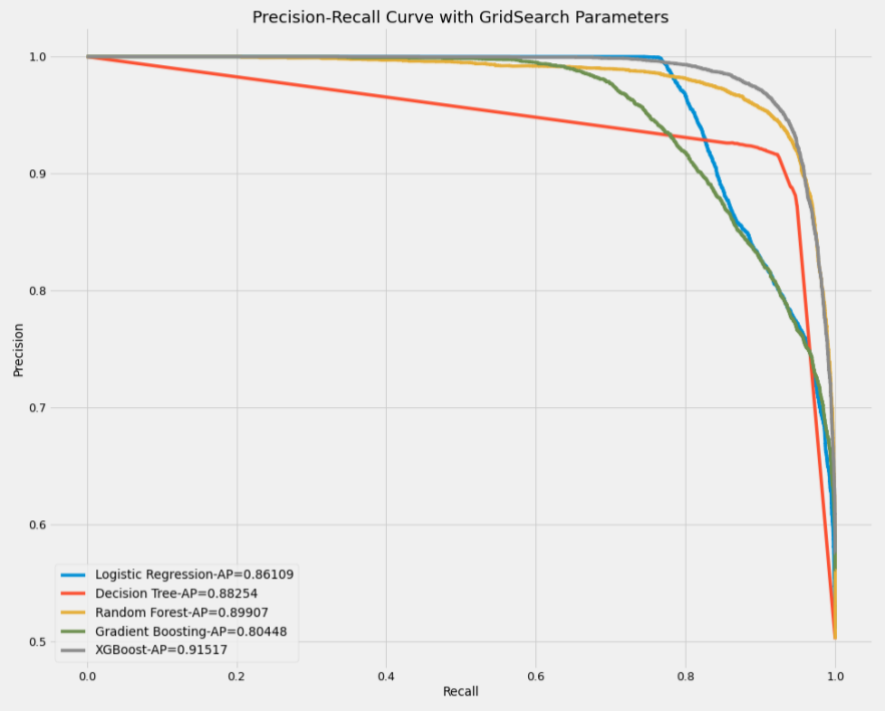
Figure 5.7 and **Error! Reference source not found.**Table 5.5 show that the XGBoost model outperforms other models in terms of decile performance. It also outperforms a random estimate consistently, with XGBoost considerably outperforming Random Forest. In terms of accuracy, memory consumption, and time-consuming, the XGBoost classifier surpasses the other classifiers. To the best of our knowledge, this is the first study on predicting employee promotion using the XGBoost classifier method.

Table 5.5 Evaluation metrics with GridSearch parameters of different classifiers.

Classifiers	Accuracy	Precision	Recall	F1-score	ROC AUC
Logistic Regression	0.88%	0.883%	0.8832%	0.88%	0.8835%
Decision Tree	0.92%	0.91%	0.91%	0.91%	0.91%
Random Forest	0.93%	0.933%	0.9328%	0.93%	0.9327%
Gradient Boosting	0.86%	0.858%	0.8582%	0.86%	0.8582%
XGBoost	0.94%	0.94%	0.94%	0.9398%	0.9397%



a)



b)

Figure 5.7 Result with GridSerach parameters of different classifiers for a) ROC b) Precision-Recall Curve.

As the result show, while random forest rely on their randomization steps to help them achieve higher generalization, this is still insufficient to prevent over-fitting in this scenario. On the other hand, XGBoost attempts to create new trees that complement the existing ones. Boosting improves training for difficult-to-classify data points. Another significant thing to consider is the over-fitting experienced by classifiers other than XGBoost, notwithstanding regularization or the addition of randomness, as the case may be. Because of its superior intrinsic regularization, XGBoost solves this issue and hence works wonderfully in our scenario.

The XGBoost classifier is also designed to be fault-tolerant in a distributed setting and is optimized for fast, parallel tree construction. The XGBoost classifier accepts DMatrix data. DMatrix is an XGBoost internal data structure that is designed for both memory economy and training speed. DMatrixes were created here by combining numpy arrays containing features and classes. Due to those reasons, the XGBoost classifier was selected as the best classifier for the dataset.

Furthermore, XGBoost surpasses the competition, and its time consumption is reasonable. As a result, the XGBoost classifier, which is based on 13 features, is chosen as the final prediction model, with an accuracy and AUC of 94.036%, a recall of 94%, and a precision of 94%. It outperformed the baseline model in terms of accuracy, increasing it by up to 94%, as shown in Figure 5.8.

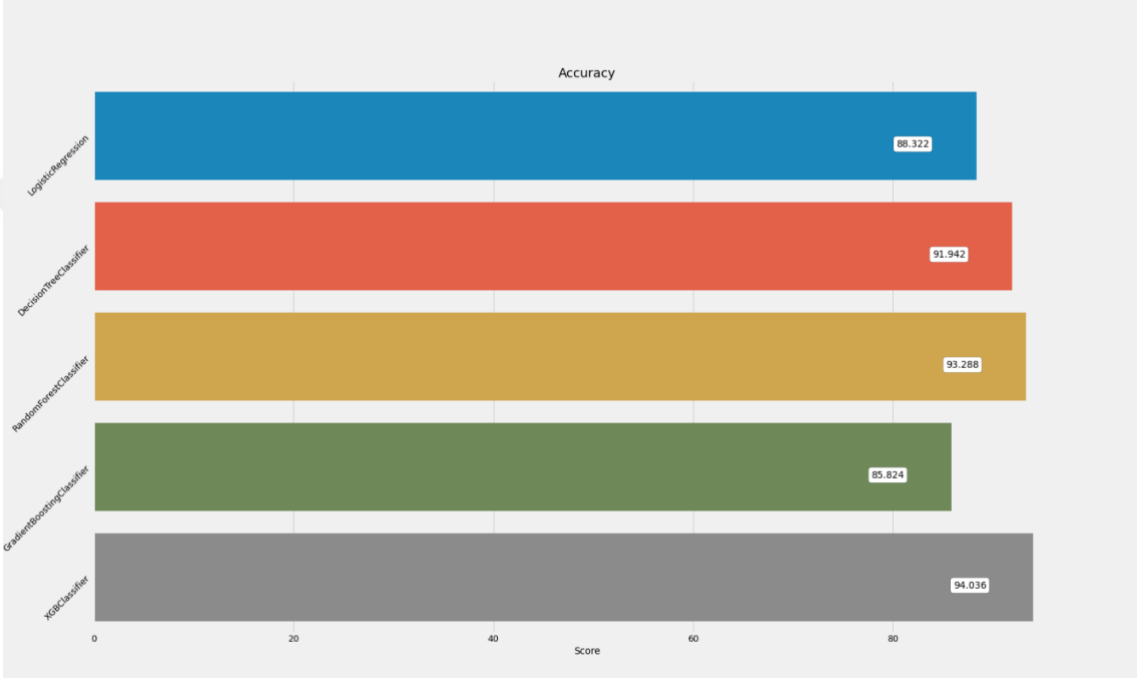


Figure 5.8 Final accuracy of the classifiers.

5. 4 Other Results

In the end, some reports and probabilities were done to understand our data.

1. Do Older Employees get more Promotion than Younger Employees?

In this phase got that younger employee got more promotions than older with 0.088% for younger and 0.071% for older.

2. What is the Probability to get Promoted if an employee has won an award?

The Probability of an Employee getting Promotion is 0.4402%, and the Probability of the employee getting promoted after winning an award is 0.076%.

3. What is the Average Training Score of those Employees who got Promotions?

The Average Training Score for the Employees who got Promotions is 71

4. What is the Impact of Gender on Promotions?

The Gender Gap in Promotion was checked and got 0.083 for males and females 0.089.

5. What is the Probability of Freshers getting Promoted?

First, we consider the employees who have worked for less than equal to two years, after that got Probability of a Fresher being Promoted is 0.0844%

6. CONCLUSION

In this thesis, we focus on the employee promotion dilemma and attempt to forecast whether an employee will be promoted at his or her present company. We define it as a binary classification issue in which employees are classified as either not being promoted or being promoted to a higher position in the organization ("promoted"). To solve this categorization problem, we employ supervised machine learning methods. The experimental outcomes are examined from multiple viewpoints and compared to various baseline models. The first findings demonstrate that our proposed models outperform the baseline models. Our approach's primary contributions are the application of machine learning algorithms, particularly XGBoost, and the development of a framework for forecasting employee promotion, which will be applied and generalized to all prediction issues, not only our problem of predicting employee promotion.

To summarize, this study is separated into five stages. Input data phase, which involves exploratory data analysis (univariate, bivariate, and multivariate) as well as data comprehension and visualization. The preprocessing phase includes many steps, such as data imputation of invalid or missing data. The data were divided into training (80%) and testing (20%) parts. This allows us to implement good engineering features to have suitable data for our problem. Preprocessing and manipulating data is another important phase of our study. Here, the SMOTE method is used to oversample our data. Feature scaling is also a strategy for normalizing the range of independent variables or data characteristics. To build our model, we trained it on a training set and verified it on a test set during the data modeling phase. XGBoost, Random Forest, Decision Tree, Logistic Regression, AdaBoost, and Gradient Boosting machine algorithms are used as classification algorithms for the prediction model. The best model is used to test various classifiers. In the last phase, the evaluation (fine and tune) phase, every occurrence of the above situation would be categorized based on whether or not the employee promotes the firm. The number of cases properly categorized by the model may be determined using a typical confusion matrix. A classification report would show the model's accuracy,

precision, recall, and F1-Score, and find the best classifier to predict whether an individual will be promoted inside the firm.

Machine learning techniques were used to discover the characteristics that may lead to an employee's promotion inside the firm and, more importantly, to forecast the chance of particular employees being promoted within the organization. Using this approach, the business may select the workers that have the best possibility of advancing the organization and then provide them with limited incentives. The XGBoost classifier generated the best results for the supplied dataset, with accuracy and ROC AUC of 0.94036%, a recall of 0.94%, and a precision of 0.94%. XGBoost is recognized as a superior algorithm in terms of memory use efficiency, accuracy, and running time. It is essentially a very robust and scalable approach for handling all types of noise from large data sets and converting the data into a suitable acceptable shape for precise outcomes. For these reasons, the XGBoost approach is suggested as a top priority for employee promotion prediction to successfully enable the business to make the optimal selection. The suggested automated predictor's results show that the important promotion factors are average training score, number of trainings, total score, and performance.

The data analysis results constitute a beginning point for the creation of increasingly efficient employee promotion classifiers. The use of extra datasets or just updating them regularly, the use of feature engineering to uncover new relevant qualities in the dataset, and the availability of more information. The use of more datasets or simply updating them regularly, the use of feature engineering to identify new significant characteristics from the dataset, and the availability of additional information on employees would improve the overall knowledge of the factors that help companies determine which employees will be promoted, thereby increasing the time available to personnel departments to assess and plan the tasks required to mitigate risk.

XGBoost is, in our opinion, the most successful business strategy. Because precision reveals the biggest errors in promotions. Furthermore, it has the highest percentage of results among other models, with a recall of 94%, that the promotion strategy may achieve its ideal aims while also increasing the motivation of those being promoted. This study may be used by an HR department to improve the efficiency of their performance and generate KPIs for promoting positions. This study also may be used by management

to estimate the likelihood of promotion, allowing managers to choose the best conditions for someone to be promoted. This initiative can also help managers reduce a person's impairment after receiving a promotion as a result of a mistake in selecting a promotion candidate. Finally, the HR department may use our final XGBoost model to feed in the record of the employee and receive a forecast of whether or not the individual should be promoted.

Aside from receiving a decent increase and promotion, today's talent has experienced various types of problems, which the project's HR executives or managers must address. This study can be expanded in the future by integrating features such as Scope of Development, Views on Workload Distribution, Career Goal Discussion, and Issues of Unhealthy Work Ethics. Organizing frequent feedback or a one-on-one interview on the organization's rules might assist HR in understanding the expectations.

We want to deploy the suggested model in real-world firms soon so that organizations may learn about employee promotion variables. The research would move in the direction of making this model a "Predictive Model" and addressing many concerns, i.e., advanced ones not predicting, but also answering the question "Who will be promoted?" but also "What can we do?" The model will improve in accuracy, scalability, and readiness for use in top IT corporations' HR departments.

As a result, we will continue to investigate other factors that have high correlations with the promotion problem in the next stage of the research. Furthermore, we will attempt to investigate more complex issues related to the promotion. For example, we can try to estimate promotion speed or investigate whether a promoted person is qualified for a higher-level role and then provide more appropriate management recommendations to businesses.

Finally, it is critical to assess the improvements that may be made to increase the performance of our model. Instead of eliminating the region column, we may divide the 32 columns into two sections: those with a higher possibility of promotion and those with a lower chance of promotion. It is also advised to investigate the use of deep learning models for forecasting promotion. A well-designed network with enough hidden layers

may enhance accuracy, but scalability and practical implementation must also be considered.

BIBLIOGRAPHY

Aarshay. November 23, 2020. “XGBOOST Parameters: XGBoost Parameter Tuning.” Analytics Vidhya. https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/.

Analyticsvidhya. n.d. “WNS-Analytics-Wizard-2018/Rank 1: Siddharth3977 at Master · Analyticsvidhya/WNS-Analytics-Wizard-2018.” GitHub. Accessed March 22, 2022. https://github.com/analyticsvidhya/wns-analytics-wizard-2018/tree/master/Rank%20120Siddharth3977.

Bandyopadhyay, Nilasha, and Anil Jadhav. 2021. “Churn Prediction of Employees Using Machine Learning Techniques.” Technical Journal / Tehnicki Glasnik 15 (1): 51–59. http://icproxy.khas.edu.tr/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=edb&AN=149158643&site=eds-live.

Brownlee, Jason. August 27, 2020. “Feature Importance and Feature Selection with XGBoost in Python.” Machine Learning Mastery. https://machinelearningmastery.com/feature-importance-and-feature-selection-with-xgboost-in-python/.

Chen, Kuan-Yu, Yu-Lun Hsu, and Chia-Chun Wu. 2012. “Num 2 Fall 2012 1 THE INTERNATIONAL JOURNAL OF ORGANIZATIONAL INNOVATION VOLUME 5 NUMBER 2 FALL 2012 Information Regarding The International Journal Of Organizational Innovation 4 IJOI.” The International Journal of Organizational Innovation. Vol. 5. http://www.iaoiusa.org.

Fallucchi, Francesca, Marco Coladangelo, Romeo Giuliano, and Ernesto William De Luca. 2020. “Predicting Employee Attrition Using Machine Learning Techniques.” Computers 9 (4): 1–17. https://doi.org/10.3390/computers9040086.

Febrina, Sindy Cahya. 2017. “Predicting Employee Performance by Leadership, Job Promotion, and Job Environmental in Banking Industry.” Jurnal Keuangan Dan Perbankan 21 (4): 641–49. https://doi.org/10.26905/jkdp.v21i4.1630.

ibrir, yasmine aya. 2022. "GitHub - ibriraya1/Forecasting-employees-promotion-based-on-the-personal-indicators-by-using-a-machine-learning-algori: Thesis Project". GitHub. https://github.com/ibriraya1/Forecasting-employees-promotion-based-on-the-personal-indicators-by-using-a-machine-learning-algori.

Jain, Praphula Kumar, Madhur Jain, and Rajendra Pamula. 2020. “Explaining and Predicting Employees’ Attrition: A Machine Learning Approach.” SN Applied Sciences 2 (4). https://doi.org/10.1007/s42452-020-2519-4.

Jain, Rachna, and Anand Nayyar. 2018. “Predicting Employee Attrition Using Xgboost Machine Learning Approach.” In Proceedings of the 2018 International Conference on System Modeling and Advancement in Research Trends, SMART 2018, 113–20. (1)Department of Computer Science and Engineering (CSE), Bharati Vidyapeeth’s

College of Engineering: Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/SYSMART.2018.8746940.

Jain, Rachna, and Anand Nayyar. 2018. “Predicting Employee Attrition Using Xgboost Machine Learning Approach.” In Proceedings of the 2018 International Conference on System Modeling and Advancement in Research Trends, SMART 2018, 113–20. (1)Department of Computer Science and Engineering (CSE), Bharati Vidyapeeth’s College of Engineering: Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/SYSMART.2018.8746940.

Jaiswal, Sonoo. n.d. “Logistic Regression in Machine Learning - Javatpoint.” www.javatpoint.com. Accessed April 9, 2022. https://www.javatpoint.com/logistic-regression-in-machine-learning.

Jaiswal, Sonoo. n.d. “Machine Learning Decision Tree Classification Algorithm - Javatpoint.” www.javatpoint.com. Accessed April 9, 2022. https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm.

Jaiswal, Sonoo. n.d. “Machine Learning Random Forest Algorithm - Javatpoint.” www.javatpoint.com. Accessed April 9, 2022. https://www.javatpoint.com/machine-learning-random-forest-algorithm.

Jantan, Hamidah, and AR Hamdan. 2010. “Applying Data Mining Classification Techniques for Employee’s Performance Prediction.” Knowledge …, 601–7. http://www.kmice.cms.net.my/ProcKMICe/KMICe2010/Paper/PG601-607.pdf.

Li, Merry Grace T., Macrina Lazo, Ariel Kelly Balan, and Joel De Goma. 2021. “Employee Performance Prediction Using Different Supervised Classifiers.” In Proceedings of the International Conference on Industrial Engineering and Operations Management, 6870–76.

Liu, J, T Wang, J Li, J Huang, F Yao, and R He. 2019. “A Data-Driven Analysis of Employee Promotion: The Role of the Position of Organization.” In Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics, 2019-Octob:4056–62. National University of Defense Technology, College of Systems Engineering: Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/SMC.2019.8914449.

Long, Yuxi, Jiamin Liu, Ming Fang, Tao Wang, and Wei Jiang. 2018. “Prediction of Employee Promotion Based on Personal Basic Features and Post Features.” In ACM International Conference Proceeding Series, 5–10. Association for Computing Machinery. https://doi.org/10.1145/3224207.3224210.

Machado, C. Sofia, and Miguel Portela. 2021. “Age and Opportunities for Promotion.” SSRN Electronic Journal, November. https://doi.org/10.2139/ssrn.2367639.

Najafi-Zangeneh, Saeed, Naser Shams-Gharneh, Ali Arjomandi-Nezhad, and Sarfaraz Hashemkhani Zolfani. 2021. “An Improved Machine Learning-Based Employees Attrition Prediction Framework with Emphasis on Feature Selection.” Mathematics 9 (11). https://doi.org/10.3390/math9111226.

Navlani, Avinash. n.d. “AdaBoost Classifier Algorithms Using Python Sklearn Tutorial.” DataCamp. Accessed April 9, 2022. https://www.datacamp.com/community/tutorials/adaboost-classifier-python.

PATER, IRENE E. DE, ANNELIES E. M. VAN VIANEN, MYRIAM N. BECHTOLDT, and UTE-CHRISTINE KLEHE. 2009. “EMPLOYEES’ CHALLENGING JOB EXPERIENCES AND SUPERVISORS’ EVALUATIONS OF PROMOTABILITY.” Personnel Psychology 62 (2): 297–325. https://doi.org/10.1111/j.1744-6570.2009.01139.x.

Punnoose, Rohit, and Pankaj Ajit. 2016. “Prediction of Employee Turnover in Organizations Using Machine Learning Algorithms.” International Journal of Advanced Research in Artificial Intelligence 5 (9). https://doi.org/10.14569/ijarai.2016.050904.

SagarDhandare. July 8, 2021. “Feature Scaling in Data Science!” Medium. DataDrivenInvestor https://medium.datadriveninvestor.com/feature-scaling-in-data-science-5b1e82492727.

Saradhi, V. Vijaya, and Girish Keshav Palshikar. 2011. “Employee Churn Prediction.” Expert Systems with Applications 38 (3): 1999–2006. https://doi.org/10.1016/j.eswa.2010.07.134.

Sarker, Ananya, S M Shamim, Md Shahiduz, Zama Mustafizur Rahman, Md Shahiduz Zama, and Mustafizur Rahman. 2018. “Employee’s Performance Analysis and Prediction Using K-Means Clustering & Decision Tree Algorithm Mawlana Bhashani Science and Technology University Employee’s Performance Analysis and Prediction Using K-Means Clustering & Decision Tree Algorithm.” Type: Double Blind Peer Reviewed International Research Journal Software & Data Engineering Global Journal of Computer Science and Technology: C. Vol. 18.

Tarbani, Nitesh. April 19, 2021. “Gradient Boosting Algorithm: How Gradient Boosting Algorithm Works.” Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/04/how-the-gradient-boosting-algorithm-works/.

Yedida, Rahul, Rahul Reddy, Rakshit Vahi, Rahul Jana, Abhilash GV, and Deepti Kulkarni. 2018. “Employee Attrition Prediction.” IJISET-International Journal of Innovative Science, Engineering & Technology 7 (9). www.ijiset.com.

CURRICULUM VITAE

Personal Information
Name and surname:
Yasmine Aya Ibrir

Academic Background
Bachelor's Degree, Branch Computer Sciences
Management Information Systems
Foreign Languages: English, French, Turkish