Principal Component Analysis and Employee Turnover Analysis based on Machine Learning

Mehedi Mahmud Student ID# 40262404
Github Link: https://github.com/mehedimahmudcse/INSE6220

Abstract—**Employee Turnover is a very perilous event for an organization in order to retain better employees who can befit the company. To obtain operational efficiency and talent manifestation for the organization has become a very critical challenge. Hence, leveraging machine learning algorithms such as Logistic Regression, Support Vector Machine (SVM) and Random Forest along with Principal Component Analysis (PCA), this study analyzes and predicts employee turnover.**

Principal Component Analysis (PCA) is a statistical approach where a case is reduced to essential features transforming from case-by-variables data table [1]. It transforms large correlated data into uncorrelated data. In this study, the high dimensional dataset including features such as satisfaction, tenure, salary etc. into lower dimensional space. The major idea is to significant variance with the data mitigating multicollinearity. Subsequently, predictive models are analyzed to get insights for predictors and turnover likelihood. SVM is employed to handle non-linear decision boundaries whereas regression serves as baseline model and finally random forest is analyzed to understand complex pattern regarding the subject matter. Afterwards, the models are evaluated based on accuracy, precision, recall, F1-score and area under the ROC-AUC curve (Receiver Operating Characteristic). This research apprehends a large company's employee turnover analysis with the value of PCA for dimensionality reduction and robust machine learning models to identify possible turnover mitigation.

Keywords—PCA, SVM, Regression Logistics, Random Forest, ROC-AUC

I. Introduction

Employee turnover is one of the most talked topics in corporate industries now-a-days. It has become more critical to retain talent as the job opportunities grow. Therefore, this work's main focal point is to append this criticality in the shade of PCA and different machine learning algorithms. The idea is to formulate an efficient analogy how to retain talents into organization. Principal Component Analysis (PCA) simplifies complex datasets while retaining trends and patterns [2]. In this process, the computational efficiency is achieved with uncorrelated data. The most important features are attributed in this analysis. The predictive models are also explored to optimize the subject matter.

In this study, PCA is used on the data set to analyze the employee turnover attributes. Next, the use of most significant machine learning algorithms enables the work to comprehend the vitality of employee turnover signifying different facets such as accuracy, recall, F1-score etc. The ROC-AUC curve is also taken into consideration to optimize the subject issue and analyze the heuristics behind the topic. This study aims to offer valuable insight to retain talents and lessen the turnover to enhance organizational stability. However, the classification results can be obtained in Google Colab notebook.

The report is organized in such a way so that the audience can get an overview by the literature review in the following sections. The section IV describes about the dataset where section V and VI discuss the results and analysis and section VII provides analytical discussion on the study. Based on the analysis the conclusion is depicted in the final portion of the report.

II. Principal Component Analysis

PCA is a mathematical principal to ordain correlated large datasets into uncorrelated smaller number of components. The originality of this is to analyze multivariate data. It is an applied form of linear algebra [3] and more commonly used on larger datasets. PCA generalizes the idea and performs feasible investigation on the components only.

A. PCA Algorithm

First, PCA identifies the new set of orthogonal coordinate axes through the data. This is achieved by the direction of maximum variance [4]. This is pointed as the first component of the data which is originally the least square lines of the best-fit data. For an instance, if the data matrix is X and then PCA is used on n * p [5] with the following steps:

1) **Regulation**: Let's analyze a matrix **X** where **n** columns are the samples (observations) and **p** are the variables of the observation. Therefore, in PCA a new matrix of uncorrelated data Y is obtained where,

$$\mathbf{Y = PX}$$

Where P to be the row vectors and X to be the column vectors, which can be interpreted as follows:

$$\mathbf{PX = (Px1\ Px2\ Px3\ \dots\ Pxn) =} \begin{bmatrix} p1.x1 & \cdots & p1.xn \\ \vdots & \ddots & \vdots \\ pm.x1 & \cdots & pm.xn \end{bmatrix}$$

$$\mathbf{= Y}$$

2) **Covariance Matrix:** In the above equation X is a matrix where n is the row vectors and p is the individual length for each row vector:

$$X = \begin{matrix} x1,1 & x1,2 & \dots & x1,p \\ \vdots & \vdots & \dots & \vdots \\ xn,1 & xn,2 & \dots & xn,p \end{matrix} \quad \text{for } Xi^{T} \in \mathrm{R}$$

Therefore, the covariance matrix can be computed as follows:

$$\mathbf{S} = \frac{1}{n-1} Y^T \mathbf{Y}$$

3) **Eigen Value Calculation:** In this calculation, the eigen vectors are pointed as the direction of the every principal component whereas the eigen values are the variance for the components.

For eigen value and eigen vector the following formula is prescribed:

$$\mathbf{AV = \lambda\, v}$$

Where A is any square matrix and λ is the corresponding eigenvalue [6].

Therefore the equation can be described as follows:

$$\mathbf{S = A \wedge \mathit{A}^T}$$

Here Λ is diagonal and non-negative which is the singular value of A.

4) **Principal Components:** Principal components analysis is the calculation of the transformed matrix Z of size n * p. Here Z represents the observed columns and Z can be given by

$$\mathbf{Z = YA}$$

III. Machine Learning Algorithms

A. Regression Logistics

The logistics regression a statistical model that is used in classification problem. Introduced by Cox (1958) [7] and developed by Hosmer and Lemenshow (2000), this demonstrates various practical aspects of logistic regression [8]. In employee turnover analysis, this model is significantly used. Regression logistics is used to do the best-fitting relationship among the class and features. Regression logistics is the logit function which relates predictors to log-odds:

$$\mathbf{logit(p)=ln(p/1-p)=\beta0+\beta1x1+\beta2x2+\cdots+\beta nxn}$$

Here, p is the probability of the positive class (Y=1), β0 is the intercept, β1,β2,…,βn are the coefficients for the predictors x1,x2,…,xn. The sigmoid function can be depicted as:

$$\mathbf{p=\sigma(z)=\frac{1}{1+e^{-z}}}$$

Here, z = β0+β1x1+β2x2+⋯+βnxn. As it is determined by a predicted class $\hat{Y}$ which applies decision threshold t (generally t=0.5)

$$\hat{Y} = \begin{cases} 1 & if\ p \geq t \\ 0 & if\ p < t \end{cases}$$

The main facility of this model that it can solve binary problem easily and for employee turnover it is a good method.

B. Support Machine Vector

The Support Machine Vector represents the better of two class for an instance employee 'staying' or 'leaving'. It is widely used in employee turnover analysis. SVM can linear and non-linear. Linear one is represented as follows:

$$w^T\ \mathbf{x + b = 0}$$

Here, w is the weight vector, b is the bias term. SVM is used for its high dimensionality, flexibility and robustness and the limitations is caused by computational complexity.

C. Random Forest

The Random Forest is an ensemble learning which is built on the basis of decision trees. It aims to find out the process of bootstrap Aggregating (bagging) and random feature selection. It takes multiple decision tree and merge them together.

For a dataset D = {(xi,yi)} where xi is the feature vector and it creates random forest M for D1, D2, ….,Dm decision trees. It shows good result on classification tasks.

Step 1: Get K records from sample N

Step 2: Build decision tree on these K records

Step 3: Choose the number of trees that are required in algorithm [10]

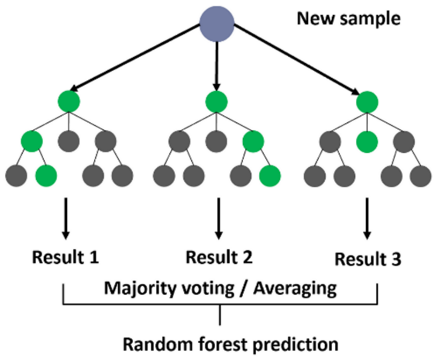


Fig 1: Random Forests.

Hence, the final class is revealed. It represents importance score on factors such as satisfaction, tenure for employee turnover. Importance score is presented as:

$$I_K = \frac{1}{M} \sum_{j=1}^{M} \Delta G_j(x_k)$$

Here $x_k$is used for splitting tree j.

IV. Data set Description

This is data is taken from large US based company [11]. The company management is very concerned of losing employees. The data consists of 10000 records for the following columns:

Department – the department of the employee

Promoted – whether the employee is promoted or not. It is denoted by 0(not promoted) 1 promoted.

Review – the last evaluation score

Projects – how many projects are done by the employee?

Salary – Salary is depicted as low, medium and high. For classification in calculation the following code is applied:

```
df =
pd.read_csv('https://raw.githubusercon
tent.com/mehedimahmudcse/INSE6220/refs
/heads/main/employee_churn_data.csv')
df['salary'] = df['salary'].replace({
    'high': 3,
    'medium': 2,
    'low': 1
})
```

Tenure: how many years the employee stayed.

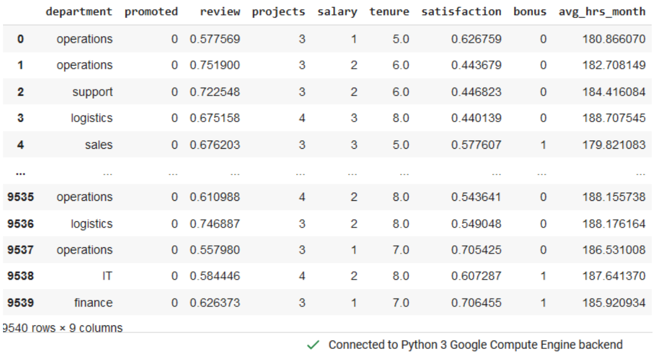
Satisfaction: it is survey for satisfaction

Bonus: 1 for bonus receive and 0 for otherwise

Avg_hrs_month: The average hours the employee works.

Left: yes for leaving and no for not.

In order to analyze the data, we load the following dataset:



	department	promoted	review	projects	salary	tenure	satisfaction	bonus	avg_hrs_month
0	operations	0	0.577569	3	1	5.0	0.626759	0	180.866070
1	operations	0	0.751900	3	2	6.0	0.443679	0	182.708149
2	support	0	0.722548	3	2	6.0	0.446823	0	184.416084
3	logistics	0	0.675158	4	3	8.0	0.440139	0	188.707545
4	sales	0	0.676203	3	3	5.0	0.577607	1	179.821083
...	...	...	...	...	...	...	...	...	...
9535	operations	0	0.610988	4	2	8.0	0.543641	0	188.155738
9536	logistics	0	0.746887	3	2	8.0	0.549048	0	188.176164
9537	operations	0	0.557980	3	1	7.0	0.705425	0	186.531008
9538	IT	0	0.584446	4	2	8.0	0.607287	1	187.641370
9539	finance	0	0.626373	3	1	7.0	0.706455	1	185.920934

9540 rows × 9 columns

✓ Connected to Python 3 Google Compute Engine backend

Fig 2: Data Load

In this work we generally looking for the analysis for employee turnover and thus we also analyze whether any duplicate data is there by the following code:

```
df.duplicated().sum()
```

It returns 0. As this is turnover analysis premier, we see the department wise employee left status:

```
import matplotlib.pyplot as plt
pd.crosstab(df.department,df.left).plo
t(kind='bar')
plt.title('Turnover Frequency for
Department')
plt.xlabel('Department')
plt.ylabel('Frequency of Turnover')
plt.savefig('department_bar_chart')
```

It generates the following output

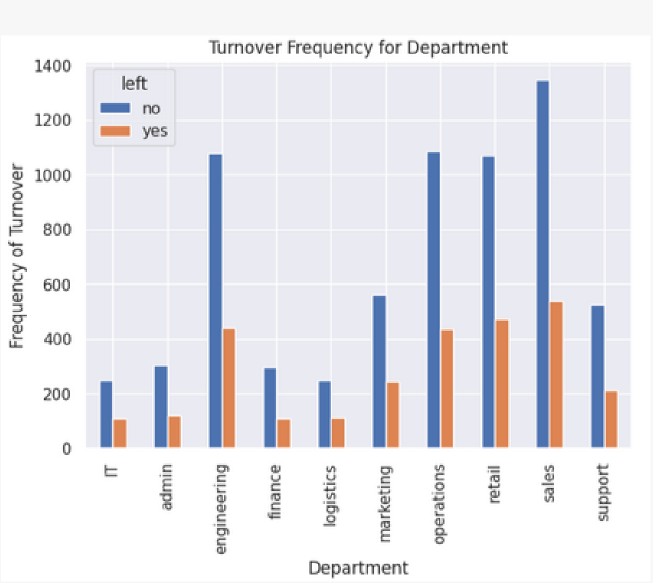


Fig 3: Employee turnover status

From the above figure it is shown that, the sales has the maximum retain and maximum leaving status.

We also analyze the salary status as per the employee as it is a node for employee turnover.

```
y =df['salary']
y.value_counts().plot(kind='pie')
plt.ylabel('')
plt.show()
```

We get the following output:

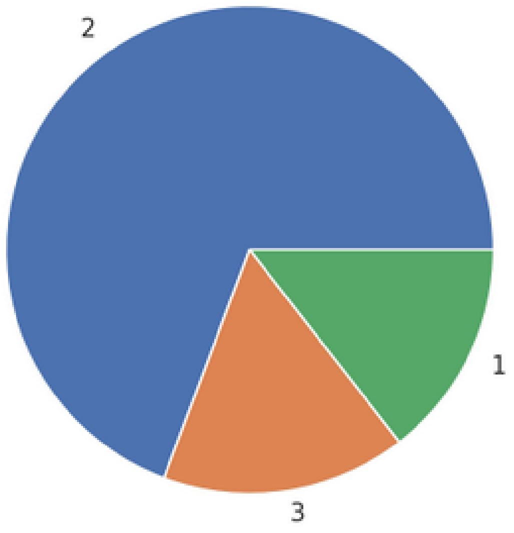


Fig 4: Employee Salary Status

From the figure, the medium salary (2) is the highest for the employees.

The employee turnover plot also analyzed for the data representation:

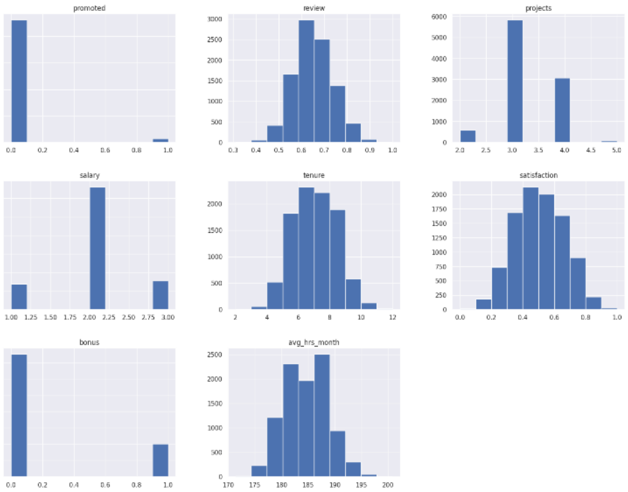


Fig 5: Employee Turnover.

The responsible features can be depicted as the Boxplot and in this dataset the following features are attributed in the event of employee turnover of the organization:

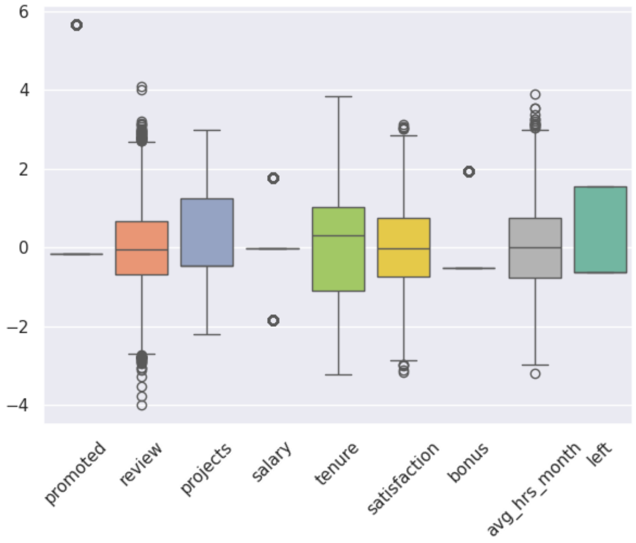
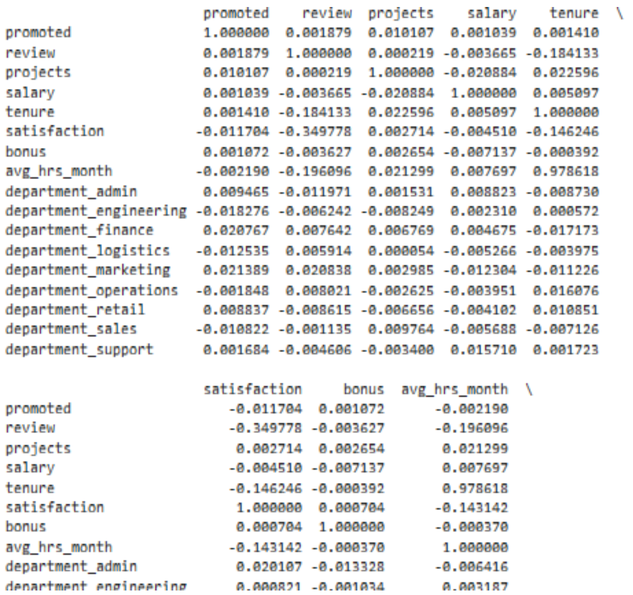


Fig 6: The Box-plot

In order to demonstrate the data set the correlation of the data has to be depicted. In the Fig 7, the positive output shows the data are highly correlated and the negative data are not. It is noted that, in case of satisfaction the negative correlation is prominent. Therefore, in order to understand the correlation, the pair plot is considered. In every numeric value in the data set as per the hue satisfaction is analyzed and thus the pair plot is demonstrated to observe the correlation. In the figure 8, the dispersed correlation is observed in case of review and satisfaction.



```
                        promoted    review  projects    salary    tenure  \
promoted                1.000000  0.001879  0.010107  0.001039  0.001410
review                  0.001879  1.000000  0.000219 -0.003665 -0.184133
projects                0.010107  0.000219  1.000000 -0.020884  0.022596
salary                  0.001039 -0.003665 -0.020884  1.000000  0.005097
tenure                  0.001410 -0.184133  0.022596  0.005097  1.000000
satisfaction           -0.011704 -0.349778  0.002714 -0.004510 -0.146246
bonus                   0.001072 -0.003627  0.002654 -0.007137 -0.000392
avg_hrs_month          -0.002190 -0.196096  0.021299  0.007697  0.978618
department_admin        0.009465 -0.011971  0.001531  0.008823 -0.008730
department_engineering -0.018276 -0.006242 -0.008249  0.002310  0.000572
department_finance      0.020767  0.007642  0.006769  0.004675 -0.017173
department_logistics   -0.012535  0.005914  0.000054 -0.005266 -0.003975
department_marketing    0.021389  0.020838  0.002985 -0.012304 -0.011226
department_operations  -0.001848  0.008021 -0.002625 -0.003951  0.016076
department_retail       0.008837 -0.008615 -0.006656 -0.004102  0.010851
department_sales       -0.010822 -0.001135  0.009764 -0.005688 -0.007126
department_support      0.001684 -0.004606 -0.003400  0.015710  0.001723

                        satisfaction     bonus  avg_hrs_month  \
promoted                   -0.011704  0.001072      -0.002190
review                     -0.349778 -0.003627      -0.196096
projects                    0.002714  0.002654       0.021299
salary                     -0.004510 -0.007137       0.007697
tenure                     -0.146246 -0.000392       0.978618
satisfaction                1.000000  0.000704      -0.143142
bonus                       0.000704  1.000000      -0.000370
avg_hrs_month              -0.143142 -0.000370       1.000000
department_admin            0.020107 -0.013328      -0.006416
department_engineering      0.000821 -0.001034       0.003187
```

Fig 7: Correlation

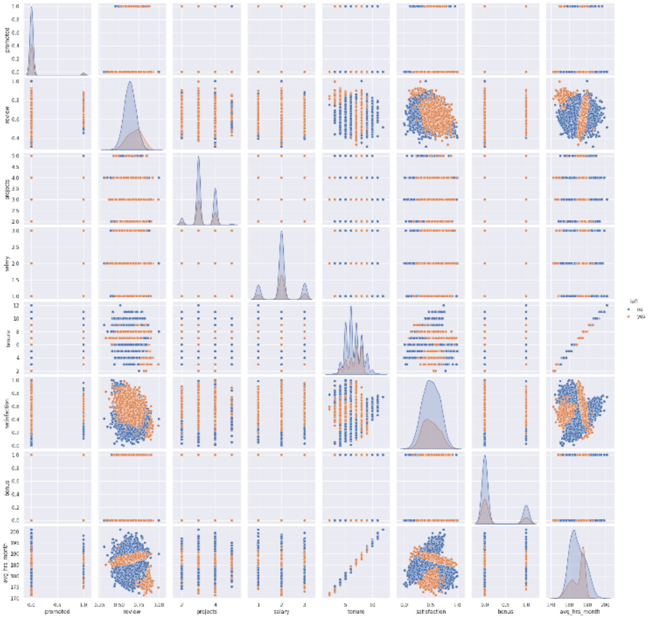


Fig 8: Pair Plot (with 'left')

V. PRINCIPAN COMPONENT ANALYSIS AND RESULT

In most cases the judged features are important and thus this research is not limited to only 5 columns and all the columns are taken into consideration. For the data the column is 10 and thus the reduced features are considered as r where r<10. Using python code the explained variances are explored. In order get the result the following code is being written. However, the components are fixed as 95%.

As we demonstrated the PCA matrix is as follows:

$$\begin{bmatrix} 0.0007 & -0.2066 & 0.0279 & 0.0089 & 0.6859 & -0.1190 & 0.0001 & 0.6869 \\ -0.0281 & -0.6858 & -0.0004 & -0.0039 & -0.0443 & 0.7249 & 0.0087 & -0.0364 \\ 0.2549 & 0.0103 & 0.6695 & -0.6434 & -0.0051 & 0.0126 & 0.2691 & -0.0087 \\ 0.8138 & -0.0241 & 0.1704 & 0.3063 & -0.0071 & 0.0151 & -0.4625 & -0.0092 \\ 0.4171 & -0.0170 & -0.2270 & 0.2778 & -0.0007 & -0.0087 & 0.8348 & -0.0010 \\ 0.3126 & -0.0201 & -0.6856 & -0.6441 & 0.0152 & -0.0076 & -0.1288 & 0.0134 \\ -0.0161 & -0.6968 & 0.0215 & -0.0072 & -0.1697 & -0.6781 & -0.0053 & -0.1583 \end{bmatrix}$$

The $\boldsymbol{\lambda}$ (lamda) is

$$\boldsymbol{\lambda} = \begin{bmatrix} 0.0000 \\ 0.1667 \\ -0.0000 \\ 0.1667 \\ 0.1667 \\ 0.1667 \\ 0.1667 \\ 0.1667 \end{bmatrix}$$

From the code the following output is being pointed out:

```
Explained Variance Ratio: [0.25773734
0.16844005 0.12807416 0.12520477
0.12468826 0.12192384]
Cumulative Explained Variance:
[0.25773734 0.42617739 0.55425155
0.67945632 0.80414458 0.92606842]
Number of components: 6
```

Fig 9: Explained Variance

From the output it is observed that r=6, i.e. the number of components is 6.

In figure 9 and figure 10 shows the Pareto-plot and scree plot respectively. These two diagram explains the amount of variance experienced by principal component. Therefore the explained variance are demonstrated as the following equation:

$$\textbf{jth explained variance ration} = \frac{\lambda_j}{\sum_{j=1}^{p} \lambda_j}; \ \mathbf{j = 1,2,3\ldots}$$

If we observe the ratio, the first component has 25.77% all the six components are significant as it these constituents more than 90%. Thus all the components are considered but in the scree plot (Fig-10) the elbow is visible in the 3rd component and the rest components are not as significant the first three components.

Z1 = 0.6859x5-1190x6+0.6869x8

Z2 = -0.6852x2+0.7249x6-0.0364x8

The component can be analyzed in Pareto plot

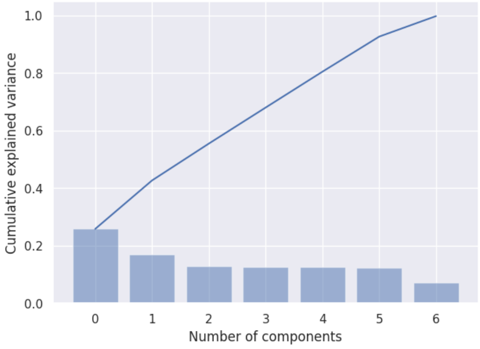


Fig: 10: Pareto Plot.

From the above figure, 5 components explain the 90% of the variance.

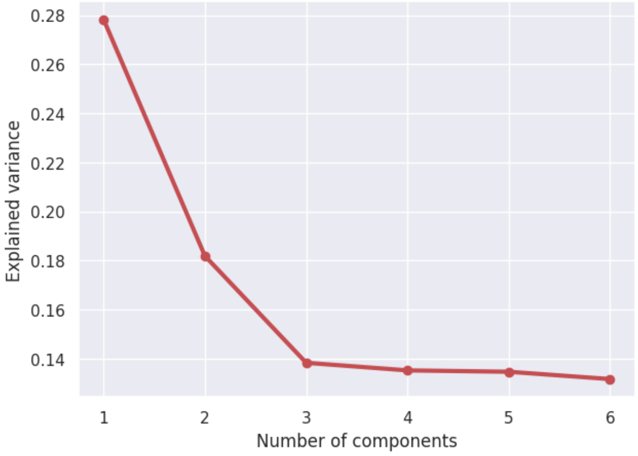


Fig 11: Scree-plot

From the scree plot it is observed that three components have significance and the other feature has less significance.

Therefore the Biplot is considered for understanding the components:

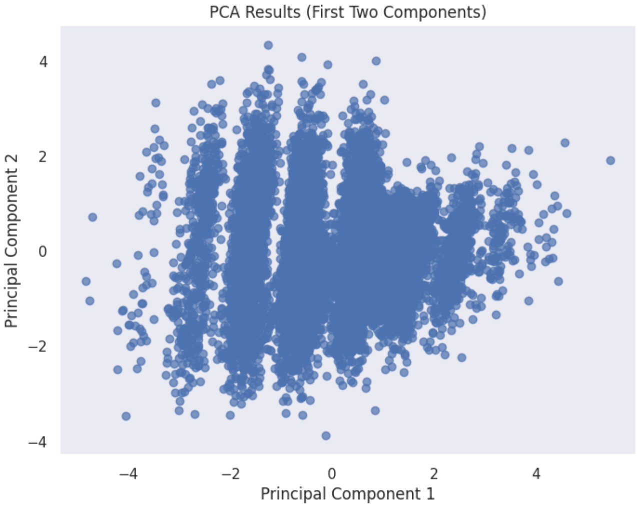


Fig 12: Bi-Plot

For the first two components demonstrates significant dispersion among the components.

VI. Classification Results

In this section the popular classification algorithm is analyzed. These algorithms are applied on the dataset and get observation using python analysis.

A. *Logistic Regression*

In this analysis we get the following confusion matrix

$$\begin{bmatrix} 852 & 478 \\ 166 & 412 \end{bmatrix}$$

The accuracy is 73.58%

B. Random Forest

In this analysis we get the following confusion matrix

$$\begin{bmatrix} 1152 & 178 \\ 117 & 461 \end{bmatrix}$$

The accuracy is 85.58%

C. SVM

In this analysis we get the following confusion matrix

$$\begin{bmatrix} 847 & 483 \\ 155 & 423 \end{bmatrix}$$

The accuracy is 66.51%

From the above analysis, random forest has the highest accuracy clearly it is better model in this context.

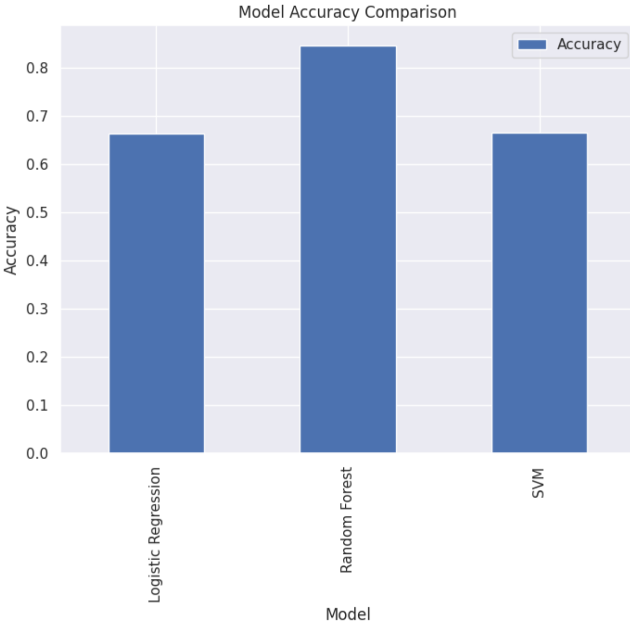
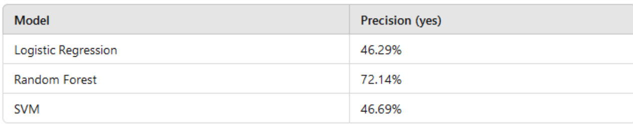


Fig 13: Model Accuracy

D. Analysis on Precision (YES)

In case of precision we get the following table:

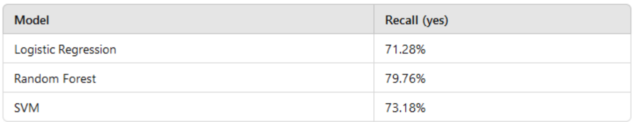


Model	Precision (yes)
Logistic Regression	46.29%
Random Forest	72.14%
SVM	46.69%

Random Forest shows a significant advantage in predicting the "yes" class, with a precision of 72.14%. This means that when the model predicts an employee will leave, it is correct 72.14% of the time.

Logistic Regression and SVM are similar, with precision scores around 46%. This means that both models are less reliable when predicting the "yes" class, as almost half of their predictions for "yes" are incorrect.

E. Analysis on Recall (YES)

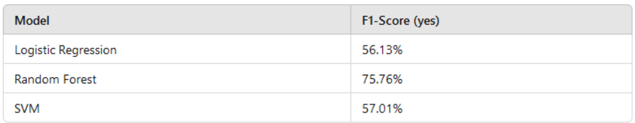


Model	Recall (yes)
Logistic Regression	71.28%
Random Forest	79.76%
SVM	73.18%

Random Forest achieves the highest recall of 79.76%, meaning it is the most successful at identifying employees who will leave. It captures nearly 80% of the true "yes" instances.

Logistic Regression and SVM have similar recall scores around 71% and 73%, respectively. These models are somewhat effective at identifying employees who will leave but miss a significant proportion of them.

F. Analysis on F1-Score (YES)

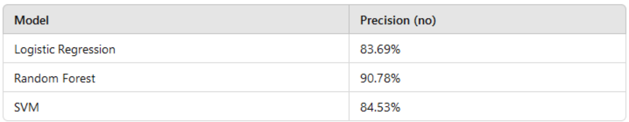


Model	F1-Score (yes)
Logistic Regression	56.13%
Random Forest	75.76%
SVM	57.01%

Random Forest again outperforms the other models with an F1-score of 75.76%, indicating a good balance of precision and recall for predicting "yes."

Logistic Regression and SVM have much lower F1-scores, around 56-57%, showing that they are less effective at both precision and recall for the "yes" class.

G. Analysis on Precision (No)



Model	Precision (no)
Logistic Regression	83.69%
Random Forest	90.78%
SVM	84.53%

Random Forest performs the best with a precision of 90.78%, meaning that when it predicts an employee will stay, it is correct 90.78% of the time.

SVM and Logistic Regression are quite similar, with precision scores around 84%. These models are quite reliable in predicting employees who stay but not as much as random Forest.

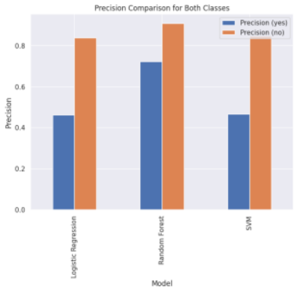
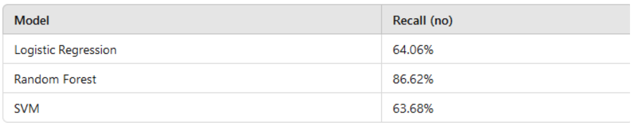


Fig 14: Precision Analysis

H. *Analysis on Recall (No)*

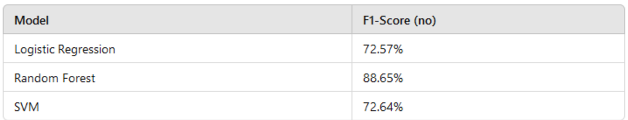


Model	Recall (no)
Logistic Regression	64.06%
Random Forest	86.62%
SVM	63.68%

Random Forest excels in recall for the "no" class with 86.62%, meaning it successfully identifies the majority of employees who stayed.

Logistic Regression and SVM have lower recall values around 64%, indicating they miss a higher percentage of the actual "no" cases.

I. *Analysis on F1-Score (No)*



Model	F1-Score (no)
Logistic Regression	72.57%
Random Forest	88.65%
SVM	72.64%

Random Forest again leads with the highest F1-score of 88.65% for the "no" class, reflecting its balanced performance in both precision and recall for predicting employees who will stay.

SVM and Logistic Regression have similar F1-scores, around 72%, showing moderate performance for the "no" class.

J. *Analysis on Confusion Matrix*

If the confusion matrices are analyzed:

For Regression Logistics:
True Negatives (TN): 852 (predicted "no" and stayed)
False Positives (FP): 478 (predicted "yes" but stayed)
False Negatives (FN): 166 (predicted "no" but left)
True Positives (TP): 412 (predicted "yes" and left)

For Random Forest:
True Negatives (TN): 1152 (predicted "no" and stayed)
False Positives (FP): 178 (predicted "yes" but stayed)
False Negatives (FN): 117 (predicted "no" but left)
True Positives (TP): 461 (predicted "yes" and left)

For SVM:
True Negatives (TN): 847 (predicted "no" and stayed)
False Positives (FP): 483 (predicted "yes" but stayed)
False Negatives (FN): 155 (predicted "no" but left)
True Positives (TP): 423 (predicted "yes" and left)

Hence from the above analysis random forest is the most significant approach to analyze the employee turnover as in case of precision, recall, F1-score and accuracy it performs better. Logistic Regression and SVM has similar performance although the accuracy of SVM is very low. Moreover, in the recall (no) it is better at identifying that employee will stay.

K. *Analysis on ROC*

This graph represents performance of the classification model at all classification threshold. In the fig-15 we see the Random Forest has an AUC of 0.92 and Logistic Regression has 0.72 and SVM has 0.60. AUC ranges from 0 to 1 and AUC =0.5 it does random guessing and when AUC is less than 0.5 it is worse than random guessing. For Random forest, The AUC of 0.92 suggests that **Random Forest** correctly classifies both "yes" (leave) and "no" (stay) cases with a high degree of accuracy across all thresholds. The model is good at separating the positive and negative classes.

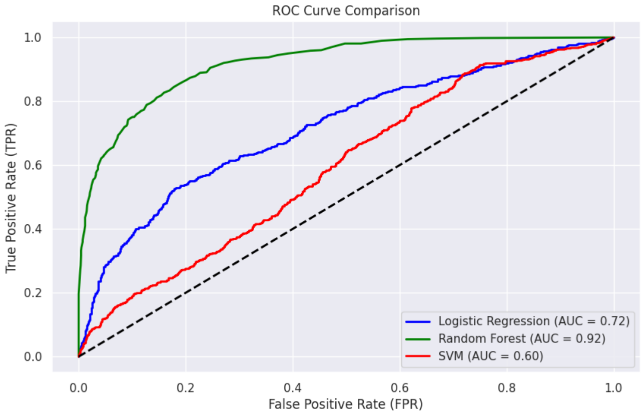


Fig-15: ROC

VII. CONCLUSION

In this study the data set is analyzed in PCA and classification models. The turnover is thus basically based on employee satisfaction and review which is observed in PCA. In order to obtain a better turnover, the study analyzes the Regression Logistic, Support Vector Machine and Random Forest. In each evaluation technique and ROC analysis, it is obtained that, the random forest is better. The confusion matrix, precision, recall and F1-score is analyzed and in all cases, random forest gives a better performance and it will constitutes the most employee retension.

VIII. REFERENCES

[1] J. Lever, M. Krzywinski, and N. Altman, "Principal component analysis," *Nature Methods*, vol. 14, pp. 641-642, 2017. doi: 10.1038/nmeth.4395.

[2] M. Richardson, "Principal Component Analysis," May 2009.

[3] J. Shlens, "A Tutorial on Principal Component Analysis," *Systems Neurobiology Laboratory*, *Salk Institute for Biological Studies*, La Jolla, CA 92037, and *Institute for Nonlinear Science*, *University of California*, *San Diego*, La Jolla, CA 92093-0402.

[4] Ait Alla and O. Rajâa, "A Review of the Literature on Employee Turnover," *American International Journal of Social Science*, vol. 8, no. 3, pp. 22, Sep. 2019. doi: 10.30845/aijss.v8n3p4.R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[5] *A. B. Hamza,* Advanced Statistical Approaches to Quality. *Unpublished*

[6] IntMath, "Eigenvalues and Eigenvectors," [Online]. Available: https://www.intmath.com/matrices-determinants/7-eigenvalues-eigenvectors.php. [Accessed: Dec. 18, 2024].

[7] *Cox, D. R. (1958). The regression analysis of binary sequences (with discussion). Journal of the Royal Statistical Society: Series B (Methodological), 20(2), 215–242. DOI: 10.1111/j.2517-6161.1958.tb00292.x*

[8] Hosmer, D. W., & Lemeshow, S. (2000). Applied Logistic Regression (2nd ed.). New York: Wiley. ISBN: 978-0471356325. DOI: 10.1002/0471722146S.

[9] Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273–297. DOI: 10.1007/BF00994018

[10] R. Yehoshua, "Random Forests," *Medium*, Mar. 24, 2023. [Online]. Available: https://medium.com/@roiyeho/random-forests-98892261dc49. [Accessed: Dec. 18, 2024].

[11] M. Stewart, "Employee Turnover," *Kaggle*, [Online]. Available: https://www.kaggle.com/datasets/marikastewart/employee-turnover?select=employee_churn_data.csv. [Accessed: Dec. 18, 2024].