CAT 2

DECISION ANALYSIS LAB

Topic:

CLASSIFICATION FOR EMPLOYEE ATTRITION

Team Members

1733011 Karthikeyan

1733019 Ragaavi D

1733024 Sanjula K R

1733026 Sri Dhanuja

1733028 Sri Hari KV

1733029 Srinandhini M

Index

1) PROBLEM STATEMENT:

Employees leave an organization when other organizations offer better opportunities than their current organizations. Continuity and sustenance and even completion of jobs are crucial issues for the companies not to suffer financial losses. Especially if the talented employees, who are at critical positions in the companies, leave the job, it becomes difficult for the organizations to maintain their businesses. Today, organizations would like to predict attrition of their employees and plan and prepare for it. However, the HR departments of organizations are not advanced enough to make such predictions in a handcrafted manner. For this reason, organizations are looking for new systems or methods that automatize the prediction of employee attrition utilizing data mining methods. This where our study comes into a bigger picture. We have used models to classify the employee attrition rate based on the given inputs from the organization. We observe that machine learning methods can be useful for predicting the employee attrition.

2) METHODOLOGY:

In this study, we use **IBM HR data** set and apply different classification methods, such as

- gradient boosting
- XG boost
- ada boost

ADD ON:

We have integrated web app UI for the user to give the inputs and get an immediate output. The steps we will go through study:

1. Data preprocessing
2. Data analysis
3. Model training
4. Model validation
5. Model predictions
6. Visualization of results

3) DATA SET:

This is a fictional data set created by IBM data scientists. Our data set consists of 35 attributes. Brief of each attributes are given below

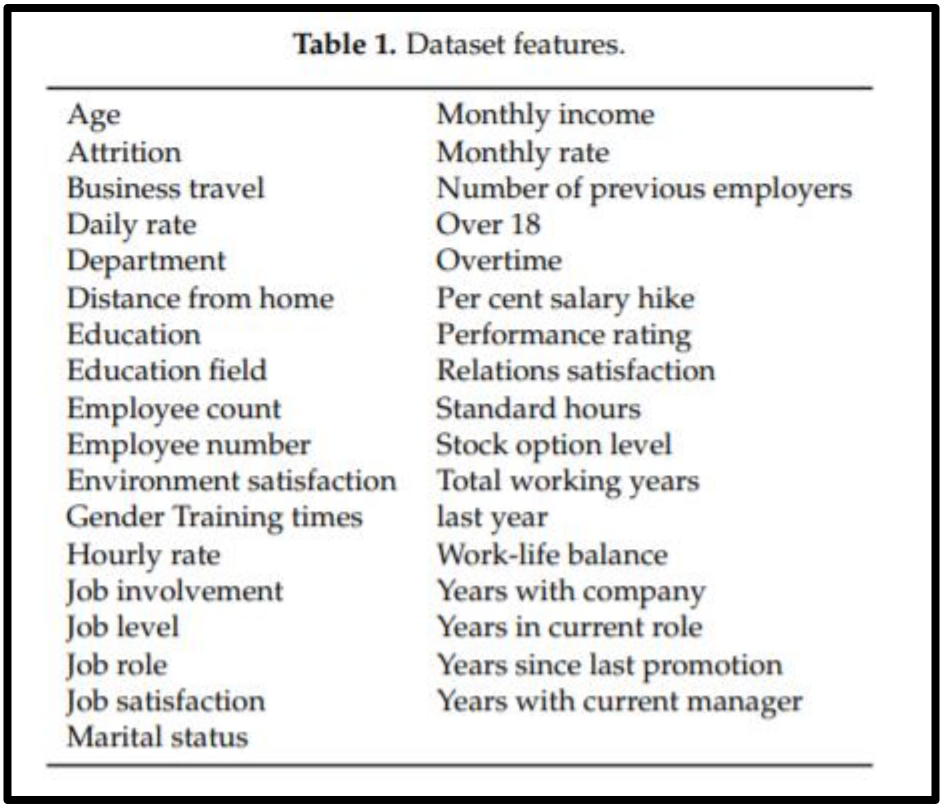


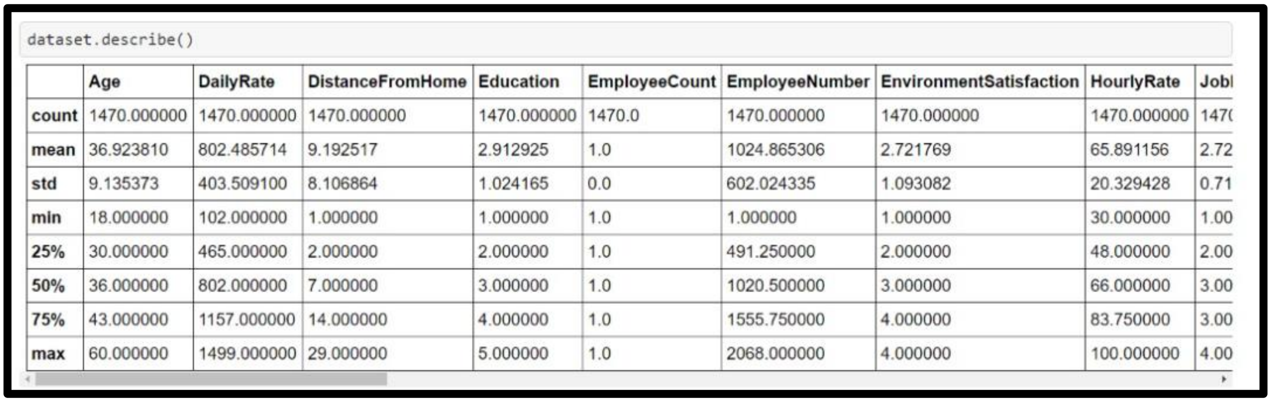
Table 1. Dataset features.

Age	Monthly income
Attrition	Monthly rate
Business travel	Number of previous employers
Daily rate	Over 18
Department	Overtime
Distance from home	Per cent salary hike
Education	Performance rating
Education field	Relations satisfaction
Employee count	Standard hours
Employee number	Stock option level
Environment satisfaction	Total working years
Gender Training times	last year
Hourly rate	Work-life balance
Job involvement	Years with company
Job level	Years in current role
Job role	Years since last promotion
Job satisfaction	Years with current manager
Marital status	

The HRM dataset used in this research work is distributed by IBM Analytics [32]. This dataset contains 35 features relating to 1500 observations and refers to U.S. data. All features are related to the employees' working life and personal characteristics. The dataset contains target feature, identified by the variable Attrition: "No" represents an employee that did not leave the company and "Yes" represents an employee that left the 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.

SUMMARY STATISTICS OF THE DATA

As shown in the output image, Statistical description of data frame (the attrition data set) was returned with the respective passed percentiles. For the columns with strings, NaN was returned for numeric operations. This analyzes both numeric and object series and also the data frame column sets of mixed data types.



```
dataset.describe()
```

	Age	DailyRate	DistanceFromHome	Education	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	HourlyRate	Jobl
count	1470.000000	1470.000000	1470.000000	1470.000000	1470.0	1470.000000	1470.000000	1470.000000	147(
mean	36.923810	802.485714	9.192517	2.912925	1.0	1024.865306	2.721769	65.891156	2.72
std	9.135373	403.509100	8.106864	1.024165	0.0	602.024335	1.093082	20.329428	0.71
min	18.000000	102.000000	1.000000	1.000000	1.0	1.000000	1.000000	30.000000	1.00
25%	30.000000	465.000000	2.000000	2.000000	1.0	491.250000	2.000000	48.000000	2.00
50%	36.000000	802.000000	7.000000	3.000000	1.0	1020.500000	3.000000	66.000000	3.00
75%	43.000000	1157.000000	14.000000	4.000000	1.0	1555.750000	4.000000	83.750000	3.00
max	60.000000	1499.000000	29.000000	5.000000	1.0	2068.000000	4.000000	100.000000	4.00

4) DATA PRE-PROCESSING:

4.1. FEATURE ENGINEERING:

Data in the real world can be extremely messy and chaotic. Hence the feature engineering plays a vital role in handling such chaotic data. Feature engineering is about **creating new input features** from your existing ones. It involves strong Data Science domain knowledge. It helps to arrive at accurate results after the model gets fitted. Feature engineering is used for following two main reasons:

- Preparing the proper input dataset, compatible with the machine learning algorithm requirements.
- Improving the performance of machine learning models.

4.2. ONE HOT ENCODING:

Feature Engineering selected here is One Hot encoding.

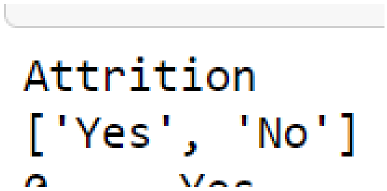
One-hot encoding is one of the most common encoding methods in machine learning. This method spreads the values in a column to multiple flag columns and assigns 0 or 1 to them. These binary values express the relationship between grouped and encoded column.

This method changes your categorical data, which is challenging to understand for algorithms, to a numerical format and enables you to group your categorical data without losing any information.

The columns encoded are:

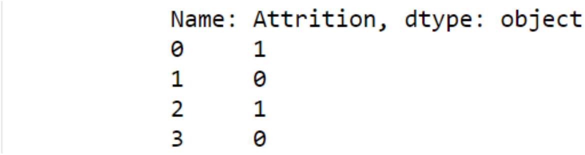
- **<u>Target Variable – Attrition (Yes/No)</u>**

Before Encoding:



```
Attrition
['Yes', 'No']
0      Yes
```

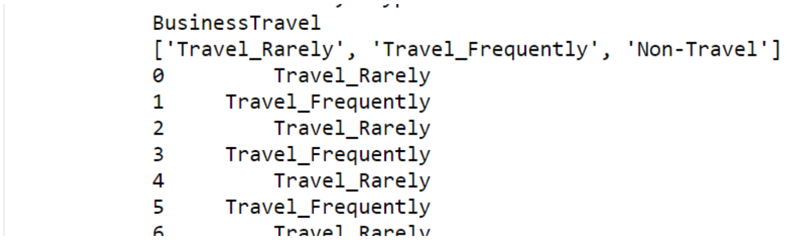
After Encoding:



```
Name: Attrition, dtype: object
0     1
1     0
2     1
3     0
```

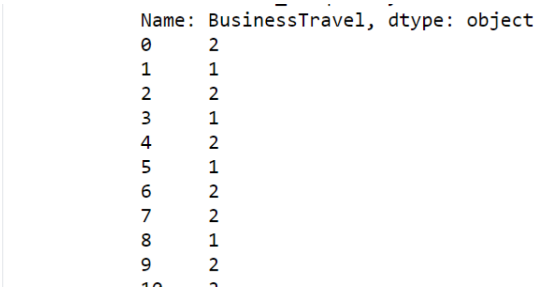
- **<u>Business Travel</u>**

Before Encoding:



```
BusinessTravel
['Travel_Rarely', 'Travel_Frequently', 'Non-Travel']
0         Travel_Rarely
1     Travel_Frequently
2         Travel_Rarely
3     Travel_Frequently
4         Travel_Rarely
5     Travel_Frequently
6         Travel_Rarely
```

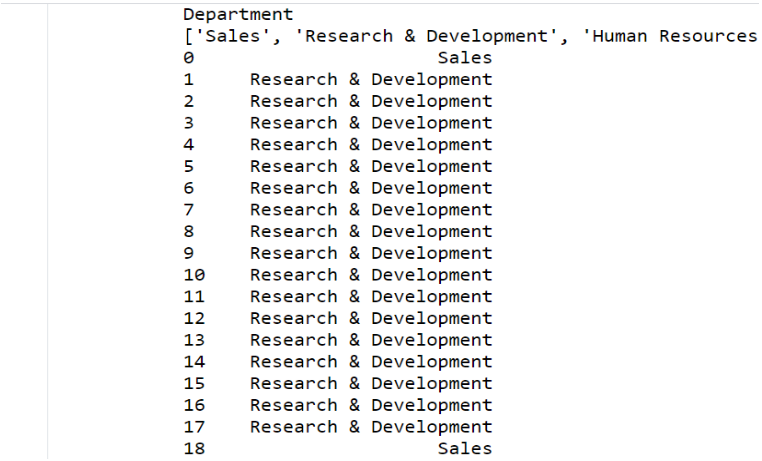
After Encoding:



```
Name: BusinessTravel, dtype: object
0     2
1     1
2     2
3     1
4     2
5     1
6     2
7     2
8     1
9     2
10    2
```

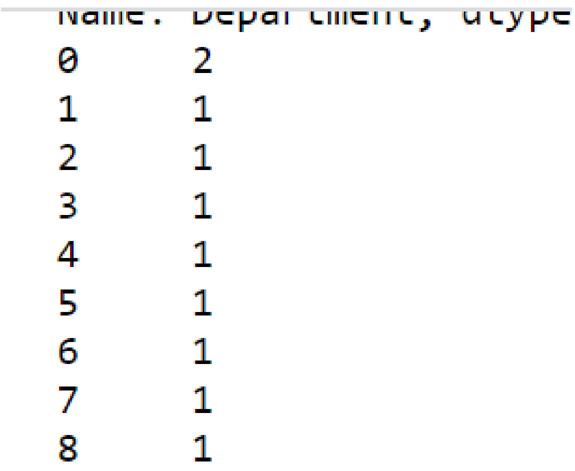
- **<u>Department</u>**

Before Encoding:



```
Department
['Sales', 'Research & Development', 'Human Resources
0                      Sales
1     Research & Development
2     Research & Development
3     Research & Development
4     Research & Development
5     Research & Development
6     Research & Development
7     Research & Development
8     Research & Development
9     Research & Development
10    Research & Development
11    Research & Development
12    Research & Development
13    Research & Development
14    Research & Development
15    Research & Development
16    Research & Development
17    Research & Development
18                     Sales
```

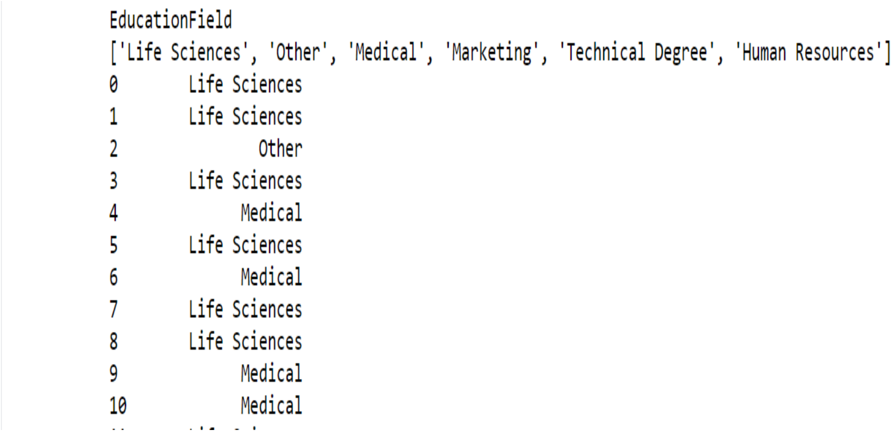
After Encoding:



```
Name: Department, dtype
0      2
1      1
2      1
3      1
4      1
5      1
6      1
7      1
8      1
```

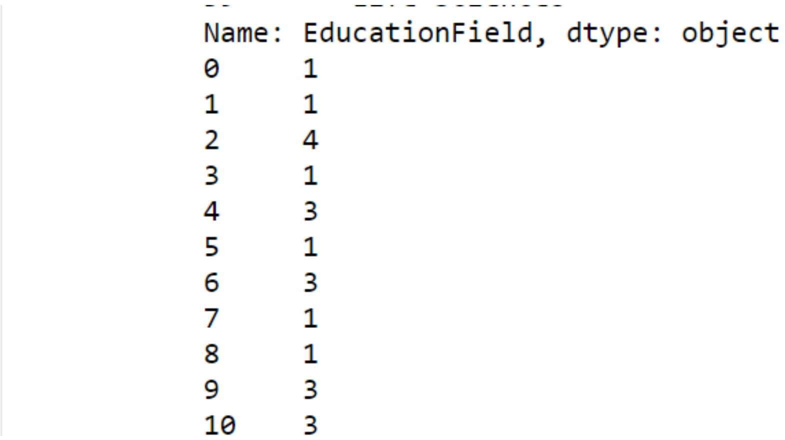
- **Education Field:**

Before Encoding:



```
EducationField
['Life Sciences', 'Other', 'Medical', 'Marketing', 'Technical Degree', 'Human Resources']
0         Life Sciences
1         Life Sciences
2                 Other
3         Life Sciences
4               Medical
5         Life Sciences
6               Medical
7         Life Sciences
8         Life Sciences
9               Medical
10              Medical
```

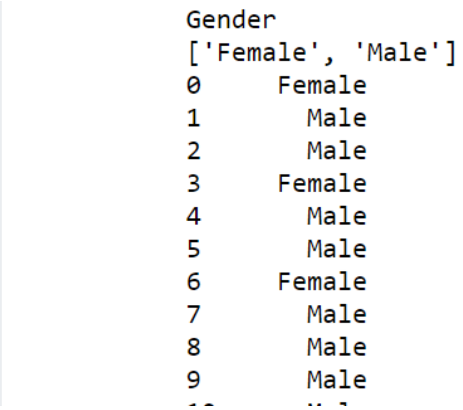
After Encoding:



```
Name: EducationField, dtype: object
0     1
1     1
2     4
3     1
4     3
5     1
6     3
7     1
8     1
9     3
10    3
```

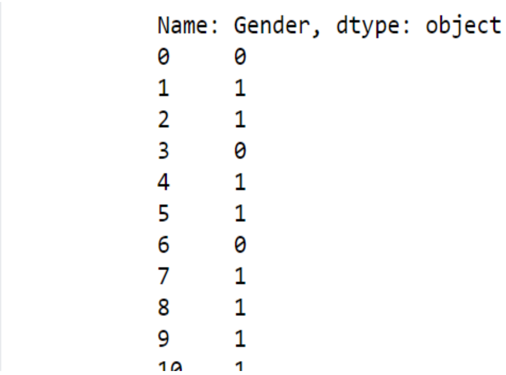
- **Gender**

Before Encoding:



```
Gender
['Female', 'Male']
0     Female
1       Male
2       Male
3     Female
4       Male
5       Male
6     Female
7       Male
8       Male
9       Male
10      Male
```

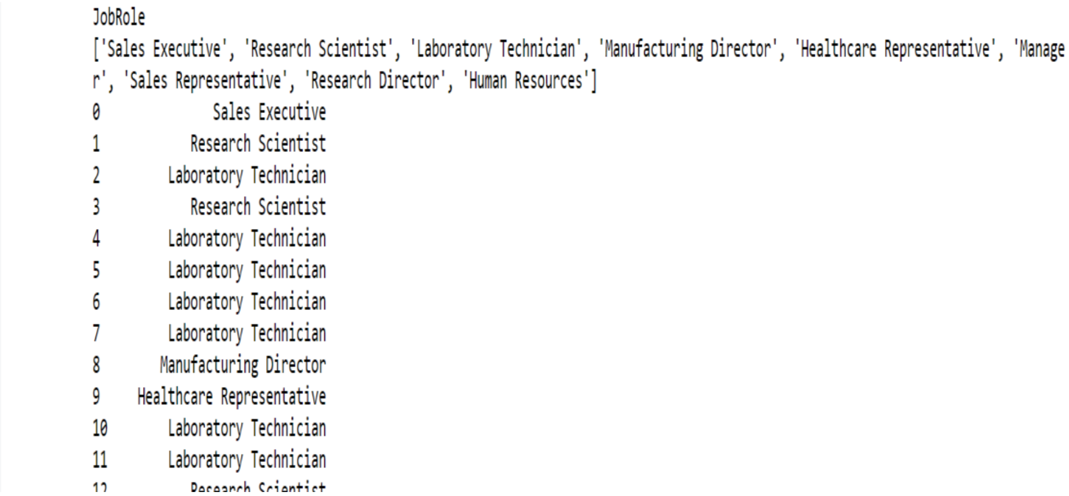
After Encoding:



```
Name: Gender, dtype: object
0     0
1     1
2     1
3     0
4     1
5     1
6     0
7     1
8     1
9     1
10    1
```

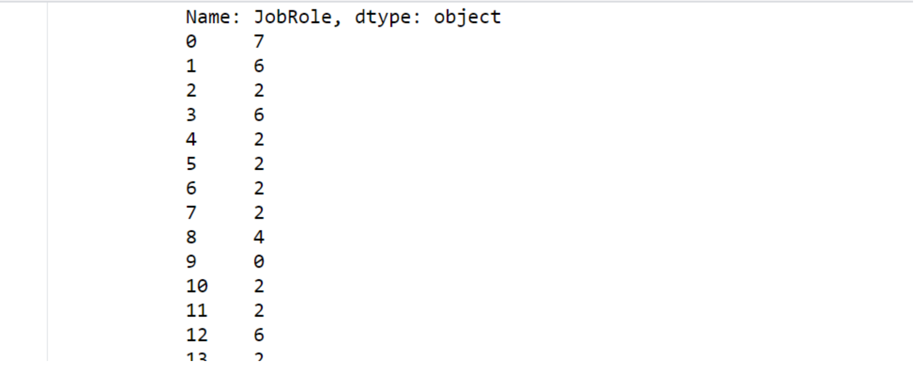
- **Job role:**

Before Encoding ;



```
JobRole
['Sales Executive', 'Research Scientist', 'Laboratory Technician', 'Manufacturing Director', 'Healthcare Representative', 'Manage
r', 'Sales Representative', 'Research Director', 'Human Resources']
0            Sales Executive
1         Research Scientist
2      Laboratory Technician
3         Research Scientist
4      Laboratory Technician
5      Laboratory Technician
6      Laboratory Technician
7      Laboratory Technician
8     Manufacturing Director
9  Healthcare Representative
10     Laboratory Technician
11     Laboratory Technician
12        Research Scientist
```

After Encoding:



```
Name: JobRole, dtype: object
0     7
1     6
2     2
3     6
4     2
5     2
6     2
7     2
8     4
9     0
10    2
11    2
12    6
13    2
```

As above 8 columns have been encoded for better output.

5) FEATURE SELECTION:

Feature Selection is the process where you automatically or manually select those features which contribute most to your prediction variable or output in which you are interested in. Having irrelevant features in your data can decrease the accuracy of the models and make your model learn based on irrelevant features.

Feature Selection helps in:

Reduces Overfitting: Less redundant data means less opportunity to make decisions based on noise.

Improves Accuracy: Less misleading data means modeling accuracy improves.

Reduces Training Time: fewer data points reduce algorithm complexity and algorithms train faster.

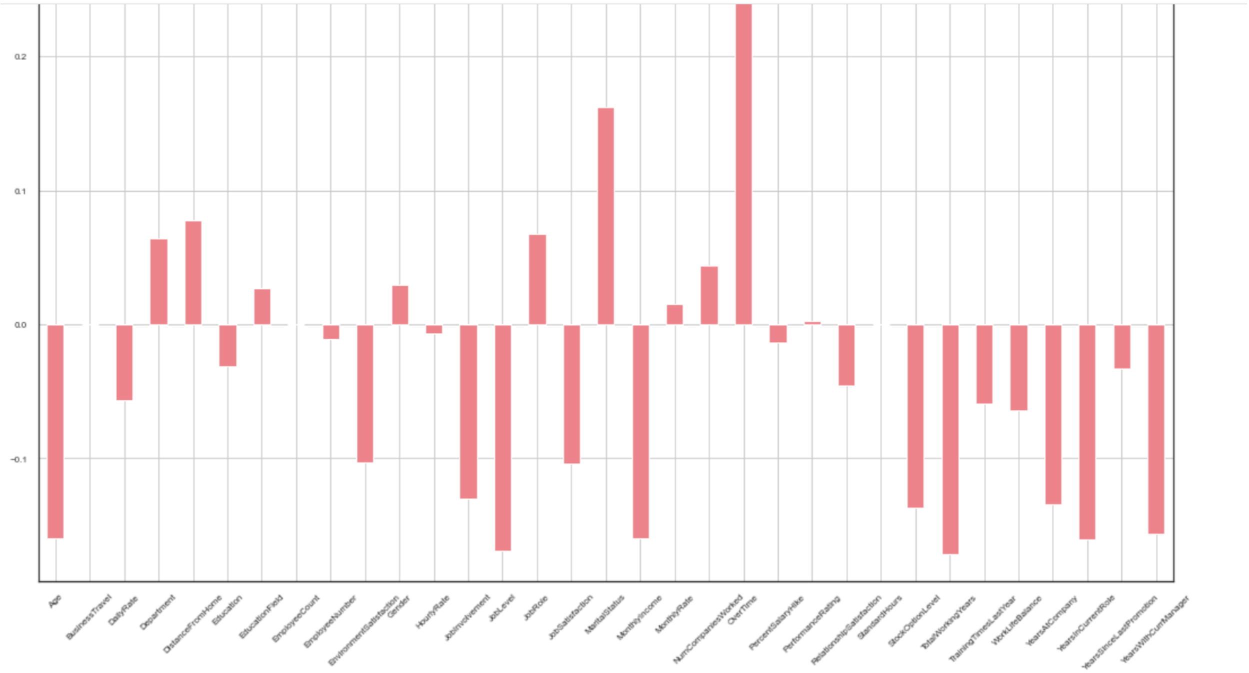
CORRELATION BASED FEATURE SELECTION:

Correlation states how the features are related to each other or the target variable.

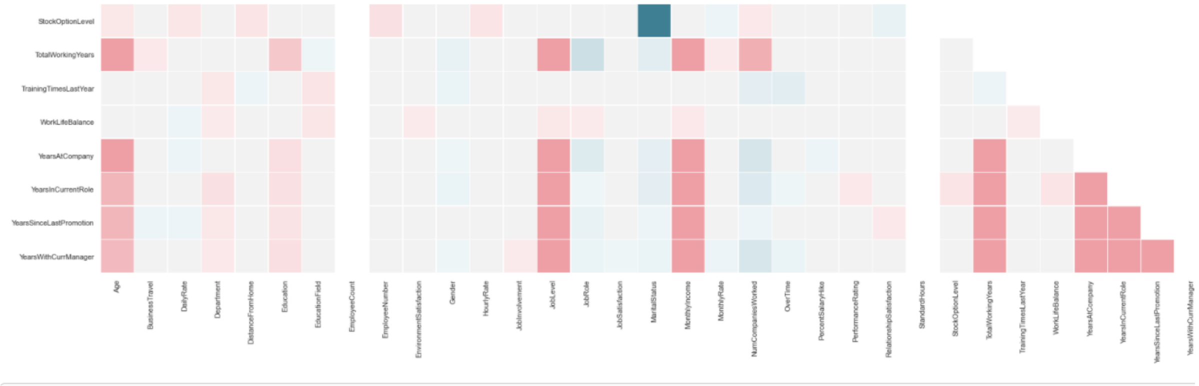
Correlation can be positive (increase in one value of feature increases the value of the target variable) or negative (increase in one value of feature decreases the value of the target variable)

Heatmap makes it easy to identify which features are most related to the target variable, we will plot heatmap of correlated features using the seaborn library.

CORRELATION GRAPH:



CORRELATION HEATMAP:



Multi collinearity has been checked using **Variable Inflation Factors (VIF).**

The value exceeding 10 indicates high multicollinearity

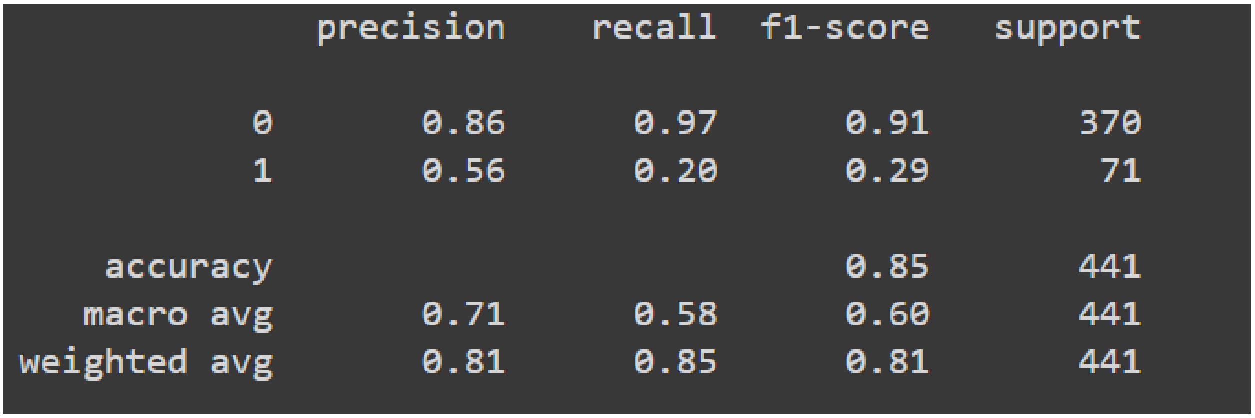
After all the above mentioned techniques been performed,

EmployeeCount, StandardHours, Over18 has been dropped.

6) MODEL STUDY:

Gradient Boosting:

Gradient boosting is a type of machine learning **boosting**. It relies on the intuition that the best possible next model, when combined with previous models, minimizes the overall prediction error. The key idea is to set the target outcomes for this next model in order to minimize the error.



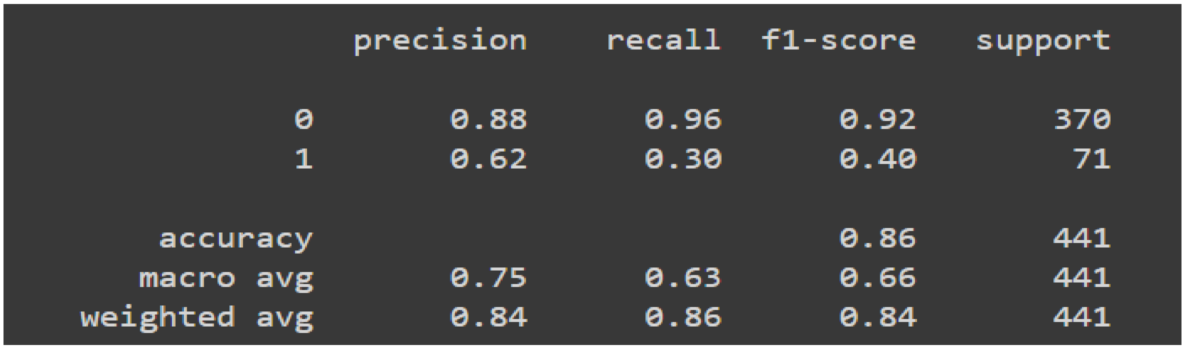
```
              precision    recall  f1-score   support

           0       0.86      0.97      0.91       370
           1       0.56      0.20      0.29        71

    accuracy                           0.85       441
   macro avg       0.71      0.58      0.60       441
weighted avg       0.81      0.85      0.81       441
```

XG Booster:

XGBoost is an implementation of gradient boosted **decision trees** designed for speed and performance. However, XGBoost improves upon the base GBM framework through systems optimization and algorithmic enhancements.



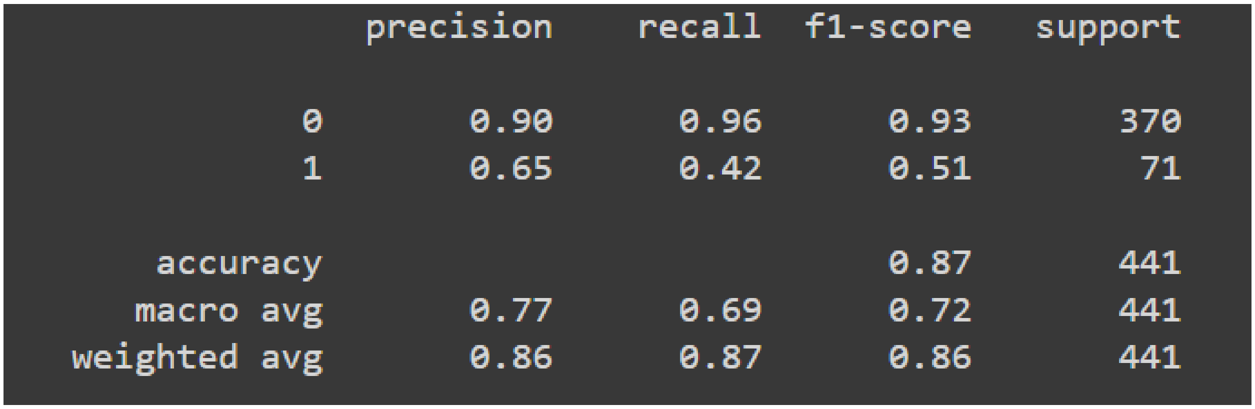
```
              precision    recall  f1-score   support

           0       0.88      0.96      0.92       370
           1       0.62      0.30      0.40        71

    accuracy                           0.86       441
   macro avg       0.75      0.63      0.66       441
weighted avg       0.84      0.86      0.84       441
```

ADA Booster:

AdaBoost algorithm, short for Adaptive Boosting, is a Boosting technique that is used as an Ensemble Method in Machine Learning. It is called Adaptive Boosting as the weights are re-assigned to each instance, with higher weights to incorrectly classified instances. Boosting is used to reduce bias as well as the variance for supervised learning. It works on the principle where learners are grown sequentially. Except for the first, each subsequent learner is grown from previously grown learners



```
              precision    recall  f1-score   support

           0       0.90      0.96      0.93       370
           1       0.65      0.42      0.51        71

    accuracy                           0.87       441
   macro avg       0.77      0.69      0.72       441
weighted avg       0.86      0.87      0.86       441
```

7) DEPLOYMENMT:

The three models get saved as .pkl file and using the pickle files Flask API was developed. **API** is a software intermediary that allows two applications to talk to each other. **Flask** is the prototype used to create instances of web application or web applications if you want to put it simple. For UI Pywebio is used, **PyWebIO** provides a series of imperative functions to obtain user input and output on the browser, turning the browser into a “rich text terminal”, and can be used to build simple web applications or browser-based GUI applications

Once the flask App is created with Pywebio UI and successfully ran locally, deployment of that application is done in Heroku. Heroku is a cloud platform as a service supporting several programming languages. One of the first cloud platforms.

For deployment in Heroku , three additional files are needed,

- Procfile - **Heroku** apps include a **Procfile** that specifies the commands that are executed by the app on startup. You can use a **Procfile** to declare a variety of process types, including: Your app's web server. Multiple types of worker processes. A singleton process, such as a clock.
- Requirements.txt - **requirements**. **txt** file is used for specifying what **python** packages are required to run the project you are looking at. Typically the **requirements**. **txt** file is located in the root directory of your project.
- Runtime.txt - **runtime**. **txt** format is case-sensitive and must not include spaces. You must also specify all three version number components (major, minor, and patch) in **runtime**. ... Whenever you change Python **runtime** versions, your dependency cache is cleared, and all dependencies need to be reinstalled.

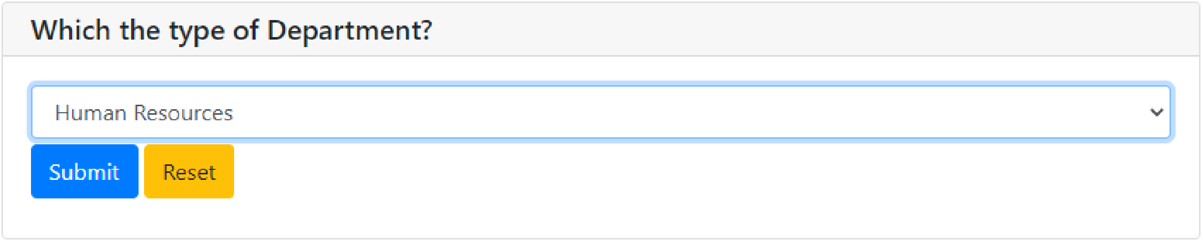
After creating all these required files entire directory will be ready for deployment. Push the directory to github by creating a separate repo for a project. Once this is done, do the following steps in Heroku

1. Create a free app in Heroku
2. Connect git with Heroku
3. Select repo of project
4. Deploy the application

Once this is done your webapp app got deployed in Heroku platform.

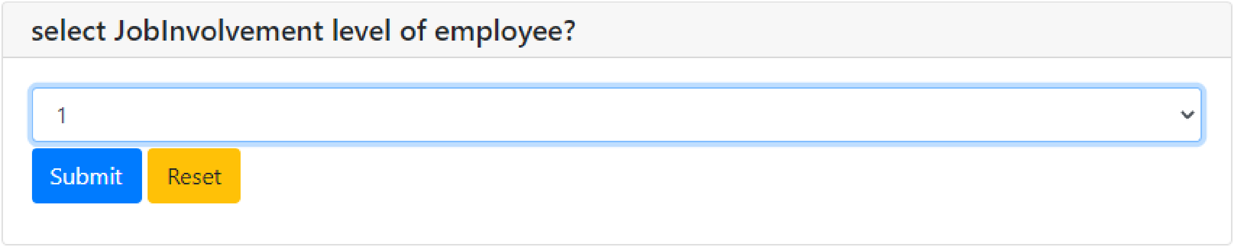
Hit the application using following URL
https://attrition-predic.herokuapp.com/

Website screenshots:



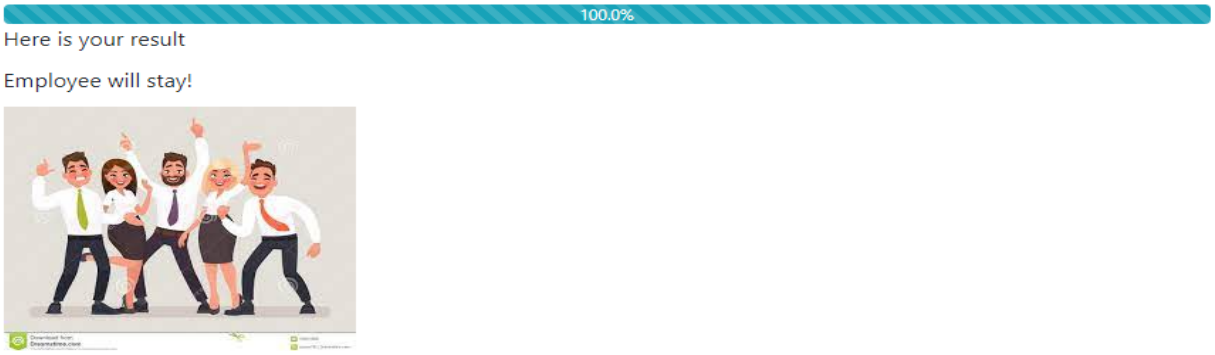
Which the type of Department?

Human Resources



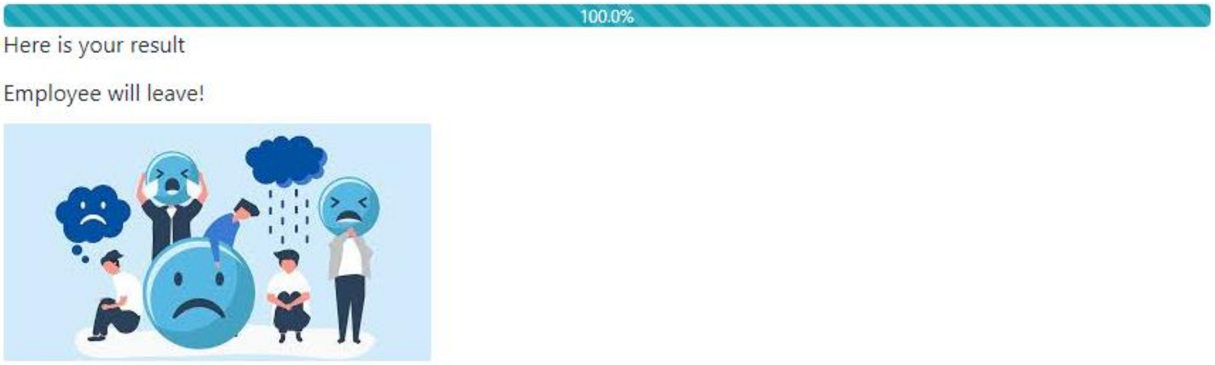
select JobInvolvement level of employee?

1



Here is your result

Employee will stay!

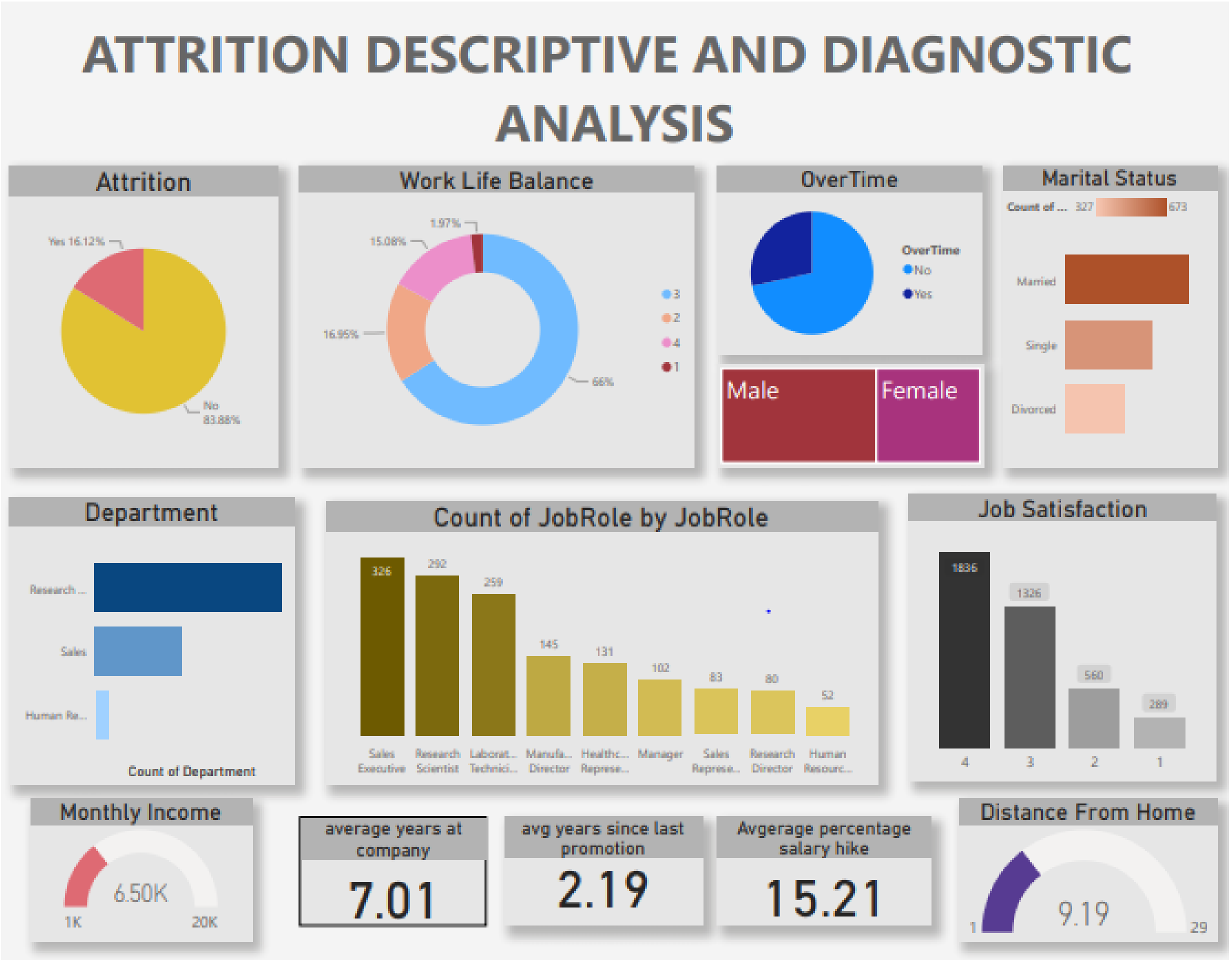


Here is your result

Employee will leave!

8) VISUALIZATION:

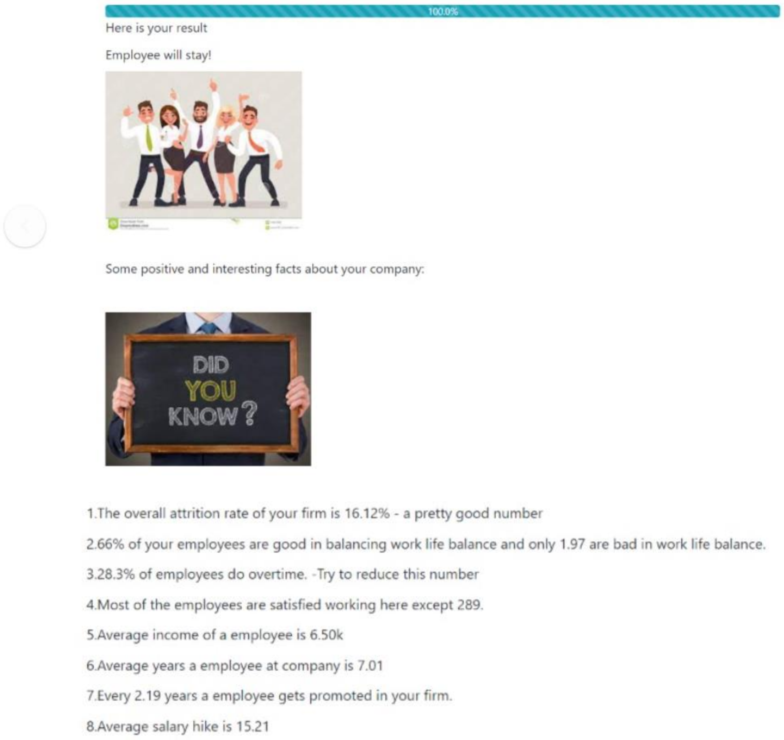
Data visualization has been done with **Power BI.**Diagnostic and descriptive analytics has been done using Power BI



9) CONCLUSION:

All four types of analytics have been implemented in our project. For prescriptive analytics we have prescribed a business strategy along with the final output. The following screenshots shows the business strategy on both the conditions [1. Chances that the employee stays in the company 2. Chances that the employee quits the company]

1. Chances that the employee stays in the company



Here is your result

Employee will stay!

Some positive and interesting facts about your company:

1.The overall attrition rate of your firm is 16.12% - a pretty good number

2.66% of your employees are good in balancing work life balance and only 1.97 are bad in work life balance.

3.28.3% of employees do overtime. -Try to reduce this number

4.Most of the employees are satisfied working here except 289.

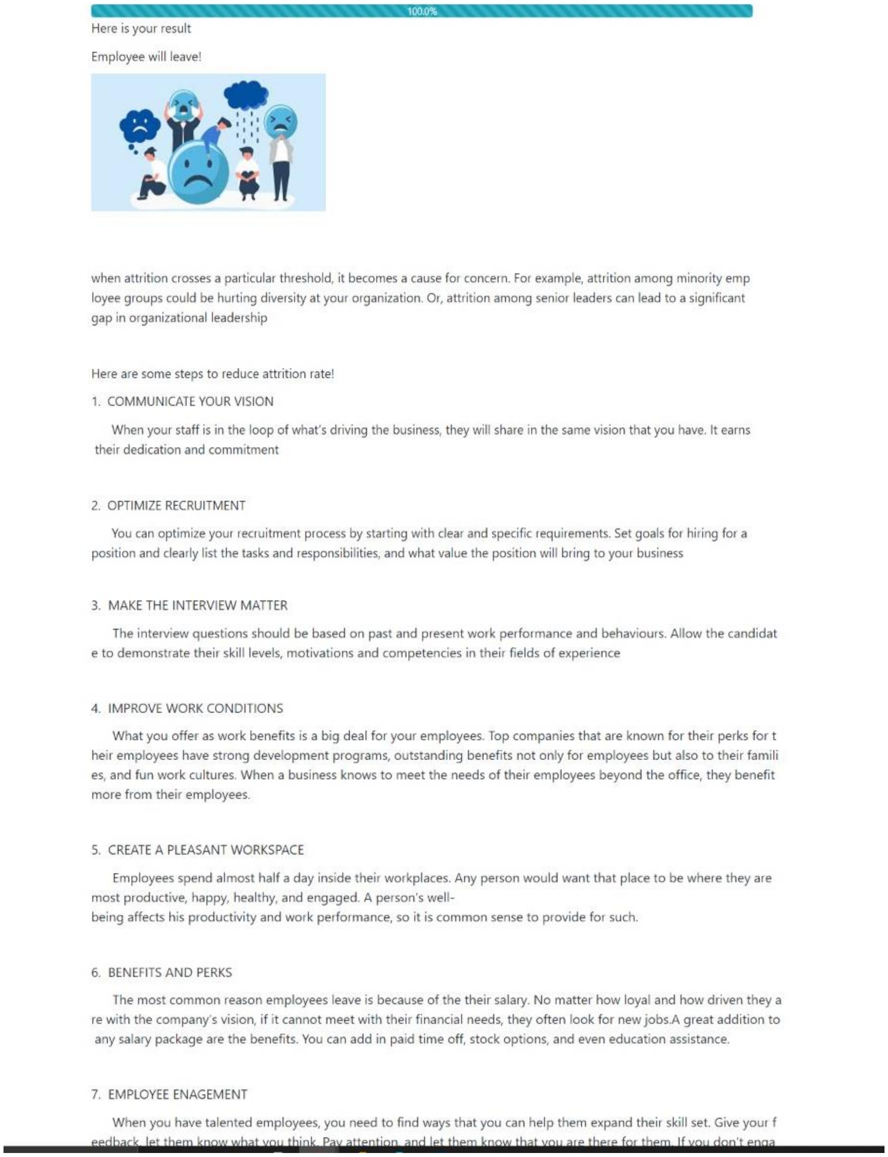
5.Average income of a employee is 6.50k

6.Average years a employee at company is 7.01

7.Every 2.19 years a employee gets promoted in your firm.

8.Average salary hike is 15.21

2. Chances that the employee quits the company



100.0%

Here is your result

Employee will leave!

when attrition crosses a particular threshold, it becomes a cause for concern. For example, attrition among minority employee groups could be hurting diversity at your organization. Or, attrition among senior leaders can lead to a significant gap in organizational leadership

Here are some steps to reduce attrition rate!

1. COMMUNICATE YOUR VISION

When your staff is in the loop of what's driving the business, they will share in the same vision that you have. It earns their dedication and commitment

2. OPTIMIZE RECRUITMENT

You can optimize your recruitment process by starting with clear and specific requirements. Set goals for hiring for a position and clearly list the tasks and responsibilities, and what value the position will bring to your business

3. MAKE THE INTERVIEW MATTER

The interview questions should be based on past and present work performance and behaviours. Allow the candidate to demonstrate their skill levels, motivations and competencies in their fields of experience

4. IMPROVE WORK CONDITIONS

What you offer as work benefits is a big deal for your employees. Top companies that are known for their perks for their employees have strong development programs, outstanding benefits not only for employees but also to their families, and fun work cultures. When a business knows to meet the needs of their employees beyond the office, they benefit more from their employees.

5. CREATE A PLEASANT WORKSPACE

Employees spend almost half a day inside their workplaces. Any person would want that place to be where they are most productive, happy, healthy, and engaged. A person's well-being affects his productivity and work performance, so it is common sense to provide for such.

6. BENEFITS AND PERKS

The most common reason employees leave is because of the their salary. No matter how loyal and how driven they are with the company's vision, if it cannot meet with their financial needs, they often look for new jobs.A great addition to any salary package are the benefits. You can add in paid time off, stock options, and even education assistance.

7. EMPLOYEE ENAGEMENT

When you have talented employees, you need to find ways that you can help them expand their skill set. Give your feedback, let them know what you think. Pay attention, and let them know that you are there for them. If you don't enga