HR ANALYTICS PROJECT- UNDERSTANDING THE ATTRITION IN HR

Problem Definition:-

Human resource analytics (HR analytics) is an area in the field of analytics that refers to applying analytic processes to the human resource department of an organization in the hope of improving employee performance and therefore getting a better return on investment. HR analytics does not just deal with gathering data on employee efficiency. Instead, it aims to provide insight into each process by gathering data and then using it to make relevant decisions about how to improve these processes

- **Attrition in HR**

Attrition in human resources refers to the gradual loss of employee's overtime. In general, relatively high attrition is problematic for companies. HR professionals often assume a leadership role in designing company compensation programs, work culture, and motivation systems that help the organization retain top employees.

HERE ARE SOME STEPS INVOLVES<

1. Loading dataset:

```
df=pd.read_csv(r'WA_Fn-UseC_-HR-Employee-Attrition.csv')
```

```
df.head()
```

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber	...	Relationship
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	1	1	...	
1	49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences	1	2	...	
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	1	4	...	
3	33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	1	5	...	
4	27	No	Travel_Rarely	591	Research & Development	2	1	Medical	1	7	...	

5 rows × 35 columns

Describing the dataset i.e.:-

Age, 'Attrition', 'BusinessTravel', 'DailyRate', 'Department', 'DistanceFromHome', 'Education', 'EducationField', 'EmployeeCount', 'EmployeeNumber', 'EnvironmentSatisfaction', 'Gender', 'HourlyRate', 'JobInvolvement', 'JobLevel', 'JobRole', 'JobSatisfaction', 'MaritalStatus', 'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked', 'Over18', 'OverTime', 'PercentSalaryHike', 'PerformanceRating', 'RelationshipSatisfaction', 'StandardHours', 'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance', 'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion', 'YearsWithCurrManager'

2.Knowing the data:

(Also known as EDA process)

```
df.describe()
```

	Age	DailyRate	DistanceFromHome	Education	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	HourlyRate	JobInvolvement
count	1470.000000	1470.000000	1470.000000	1470.000000	1470.0	1470.000000	1470.000000	1470.000000	1470.000000
mean	36.923810	802.485714	9.192517	2.912925	1.0	1024.865306	2.721769	65.891156	2.729932
std	9.135373	403.509100	8.106864	1.024165	0.0	602.024335	1.093082	20.329428	0.711561
min	18.000000	102.000000	1.000000	1.000000	1.0	1.000000	1.000000	30.000000	1.000000
25%	30.000000	465.000000	2.000000	2.000000	1.0	491.250000	2.000000	48.000000	2.000000
50%	36.000000	802.000000	7.000000	3.000000	1.0	1020.500000	3.000000	66.000000	3.000000
75%	43.000000	1157.000000	14.000000	4.000000	1.0	1555.750000	4.000000	83.750000	3.000000
max	60.000000	1499.000000	29.000000	5.000000	1.0	2068.000000	4.000000	100.000000	4.000000

8 rows × 26 columns

Now in order to know the correlation I have plot heatmap from where I have got some findings

Finding of correlation matrix

1. JobLevel is highly correlated to Age as expected as Aged employees will generally te d to occupy higher positions in the company
2. MonthlyIncome is highly correlated JobLevel as expected senior will definitely earn more
3. PerformanceRating is highly related to PercentSalaryHike which is quite obvious
4. TotalWorkingYears is also highly correlated to JobLevel
5. YearsWithCurrManager is highly related to YearsAtCompany
6. YearsAtCompany is related to YearsInCurrentRole

From above analysis I have to drop some columns which are:-

{'BusinessTravel','DailyRate','EmployeeCount','EmployeeNumber','HourlyRate','MonthlyRate' ,'NumCompaniesWorked','Over18','StandardHours', 'StockOptionLevel', 'TrainingTimesLastYear'}

Now the column transform by the label encoder are:-

```
#feature Encoding
def transform(feature):
    le=LabelEncoder()
    df[feature]=le.fit_transform(df[feature])
    print(le.classes_)
```

```
cat_df=df.select_dtypes(include='object')
cat_df.columns
```

```
Index(['Attrition', 'Department', 'EducationField', 'Gender', 'JobRole',
       'MaritalStatus', 'OverTime'],
      dtype='object')
```

And the selecting the dtypes columns and and transform the columns also done in the EDA process

3. **Split the data into training and testing**
   Now split the data in test and train by using 75:25 ratio and finding the best random state which is I assume is
   And before the I have split also the target column i.e. ATTRITION and rest into x and y respectively.

```
#feature scaling
sc=StandardScaler()
sc_df=sc.fit_transform(df.drop('Attrition',axis=1))
X=sc_df
Y=df['Attrition']
```

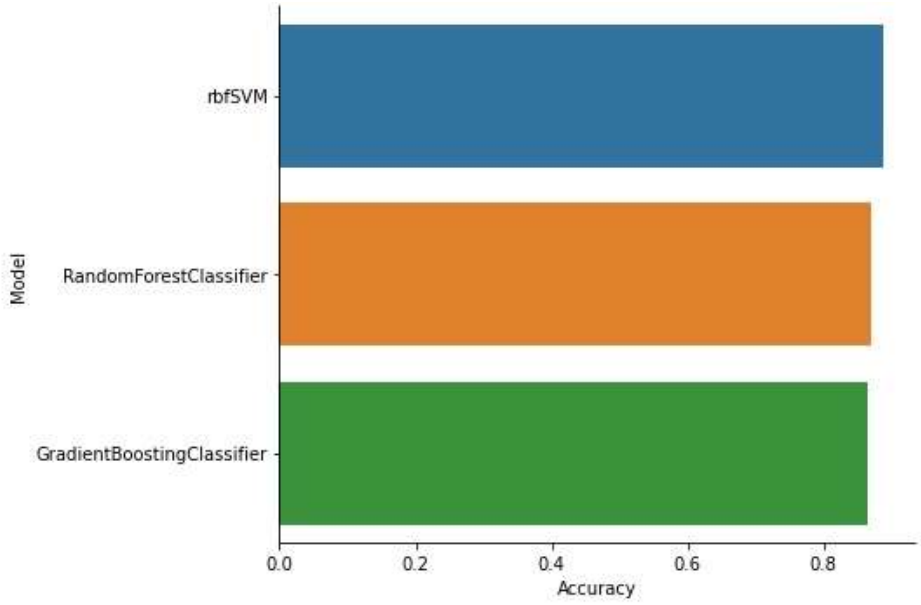
splitting the data test and training

```
x_train,x_test,y_train,y_test=train_test_split(X,Y,test_size=0.25,random_state=42)
```

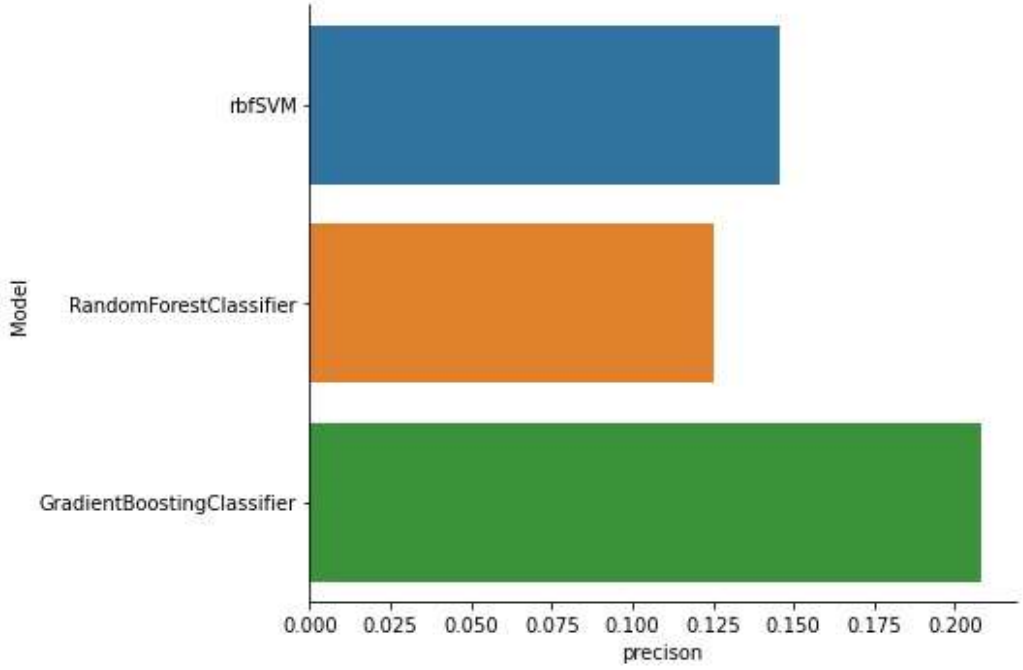
4. **Applying the machine learning model:**
   I have made class name compare in which I have passed the my models which are 'rbfSVM','RandomForestClassifier','GradientBoostingClassifier' and from which have also get some accuracy score and precision score and area under the curve. Here are some of my findings.
   In table given below:

S.no	Model name	Accuracy	Precision	Recall	Area under curve
1.	Support vector machine(rbfSVM)	0.888587	0.145833	1.00	0.943213
2.	Random forest classifier	0.869565	0.104167	0.5000	0.689944
3.	Gradient boosting Classifier	0.864130	0.208333	0.4545	0.672359

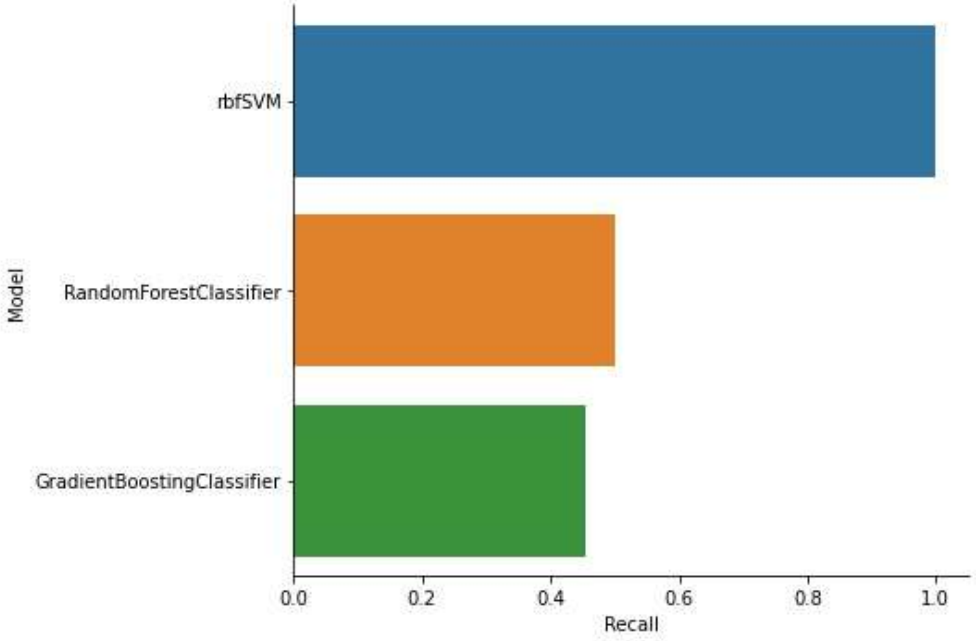
- Now the accuracy score of my models are in the form of plot



- And here is the Precision score plot



- And the recall score plot is :-



5. Conclusion

- We tried several models to get maximum accuracy. We used the **Support vector classifier**, which gives an accuracy of 88.8%.
- We used the **Random forest classifier**, which gives an accuracy of 86.95%.
- We used the **Gradient boosting classifier**, which gives an accuracy of 86.41%.

Source code/link to github: