ANALYSIS REPORT ON EMPOLYEE ATRRITION CASE IN COMPANY X USING PYTHON AND TABLEAU

BY BAKARE-BOLAJI MOYOSOREOLUWA

Problem Statement/Description

- **Description:**
- The data is for company X which is trying to control attrition. There are two sets of data: "Existing employees" and "Employees who have left". Following attributes are available for every employee.
- Satisfaction Level
- Last evaluation
- Number of projects
- Average monthly hours
- Time spent at the company
- Whether they have had a work accident
- Whether they have had a promotion in the last 5 years
- Departments (column sales)
- Salary
- Whether the employee has left

Problem Statement /Description

- Objective

1.What type of employees are leaving? Determine which employees are prone to leave next. Present your results in the presentation sheet's presentation area.

- Expected Output

1. EXPLAIN WHAT TYPE OF EMPLOYEE ARE PRONE TO LEAVE THE COMPANY.

2.PREDICT THE FUTURE EMPLOYEE WHO WOULD TEND TO LEAVE THE COMPANY.

OBSERVATIVE/EXPLORATORY Analysis

we do some Exploratory Data Analysis, where we summarize characteristics of data such as pattern, trends, outliers, and hypothesis testing using descriptive statistics and visualization.

```
#import modules
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

excelfile = pd.ExcelFile('real.xlsx')# the employee data sheet has been merge(existing employee and employee who left)
data = excelfile.parse('Sheet1')
print(data.head())
print(data.tail())
print(data.describe())
```

```
   Emp ID  satisfaction_level  last_evaluation  number_project  \
0       1                0.38             0.53               2
1       2                0.80             0.86               5
2       3                0.11             0.88               7
3       4                0.72             0.87               5
4       5                0.37             0.52               2

   average_montly_hours  time_spend_company  Work_accident  \
0                   157                   3              0
1                   262                   6              0
2                   272                   4              0
3                   223                   5              0
4                   159                   3              0

   promotion_last_5years   dept  salary  left
0                      0  sales     low     0
1                      0  sales  medium     0
2                      0  sales  medium     0
3                      0  sales     low     0
4                      0  sales     low     0
```

```
              Emp ID  satisfaction_level  last_evaluation  number_project  \
count    3571.000000         3571.000000      3571.000000     3571.000000
mean     6500.439653            0.440098         0.718113        3.855503
std      6266.484705            0.263933         0.197673        1.818165
min         1.000000            0.090000         0.450000        2.000000
25%       893.500000            0.130000         0.520000        2.000000
50%      1786.000000            0.410000         0.790000        4.000000
75%     12678.500000            0.730000         0.900000        6.000000
max     14999.000000            0.920000         1.000000        7.000000

       average_montly_hours  time_spend_company  Work_accident  \
count           3571.000000         3571.000000    3571.000000
mean             207.419210            3.876505       0.047326
std               61.202825            0.977698       0.212364
min              126.000000            2.000000       0.000000
25%              146.000000            3.000000       0.000000
50%              224.000000            4.000000       0.000000
75%              262.000000            5.000000       0.000000
max              310.000000            6.000000       1.000000

       promotion_last_5years    left
count            3571.000000  3571.0
mean                0.005321     0.0
std                 0.072759     0.0
min                 0.000000     0.0
25%                 0.000000     0.0
50%                 0.000000     0.0
75%                 0.000000     0.0
max                 1.000000     0.0
```

INSIGHT ON DATA

- we have two types of employee one who stayed and another who left the company. So, we divided them into two groups and compare their characteristics.

```
print(data.left.value_counts())
left = data.groupby('left')
left.mean()
```

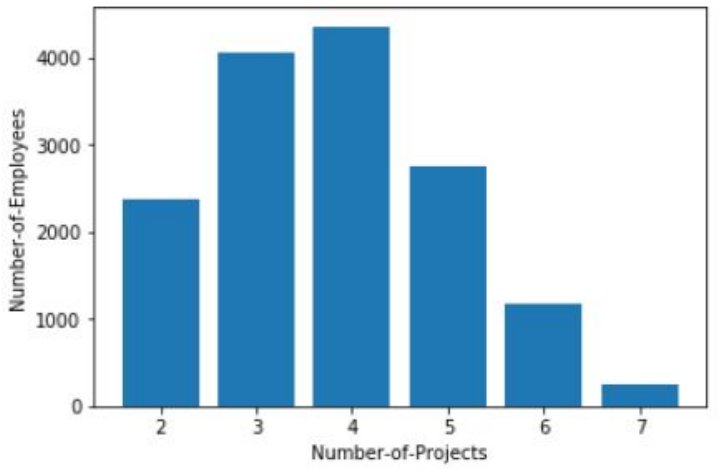
```
1    11428
0     3571
Name: left, dtype: int64
```

left	Emp ID	satisfaction_level	last_evaluation	number_project	average_montly_hours	time_spend_company	Work_accident	promotion_last_5years
0	6500.439653	0.440098	0.718113	3.855503	207.419210	3.876505	0.047326	0.005321
1	7812.340742	0.666810	0.715473	3.786664	199.060203	3.380032	0.175009	0.026251

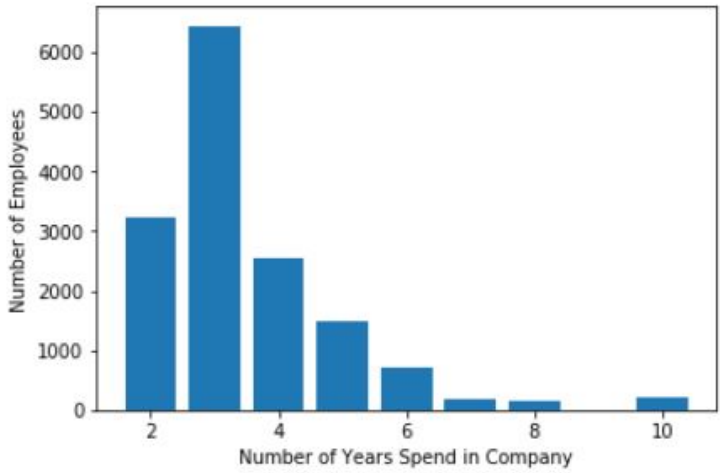
We can see that Employees who left the company had low satisfaction level, low promotion rate in the last 5 years, and worked more compare to who are still with the company. I believe we have found a little bit of clue why they left.

They say picture says a thousand words, so lets **VISUALIZE THE DATA**

```
num_projects=data.groupby('number_project').count()
plt.bar(num_projects.index.values, num_projects['satisfaction_level'])
plt.xlabel('Number-of-Projects')
plt.ylabel('Number-of-Employees')
plt.show()
```



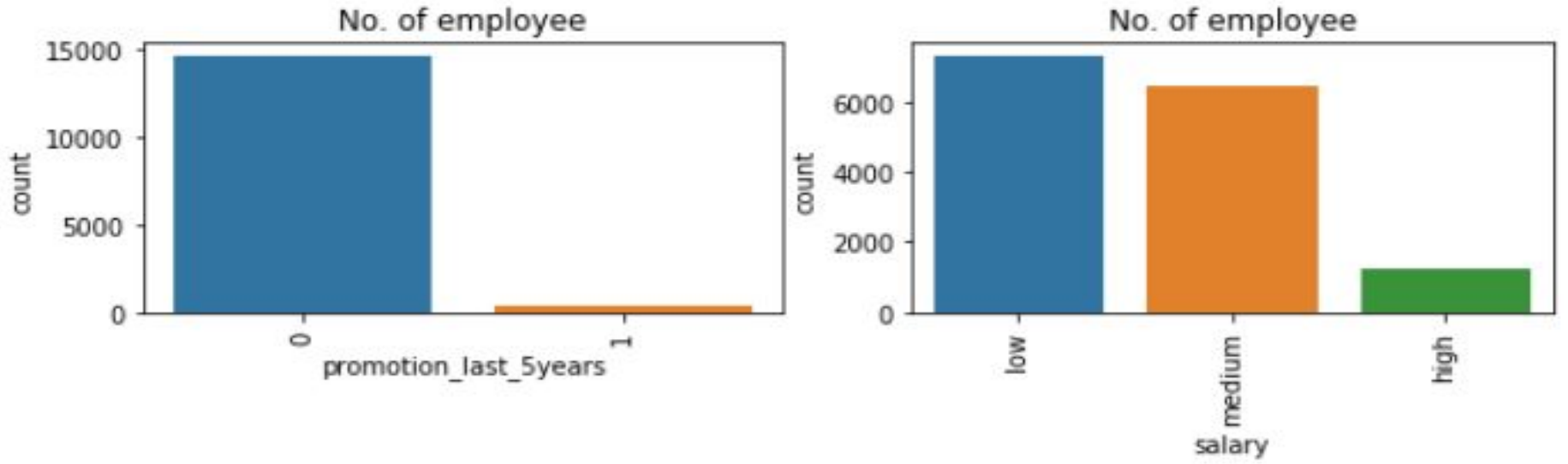

```
time_spent=data.groupby('time_spend_company').count()
plt.bar(time_spent.index.values, time_spent['satisfaction_level'])
plt.xlabel('Number-of-Years-Spend-in-Company')
plt.ylabel('Number of Employees')
plt.show()
```



- So we can see all the employees were involved in at least 3-4 p
- Lets check the time they spent in the company.
- So we can see that they spent at least 3 years in the company.
- And wow what huge gap between 3-4 years. They begin to leave, in year 4. That when they get dissatisfied with the job
- Because maybe they are over worked.

- Lets visualize there promotion rate in the last 5 years.

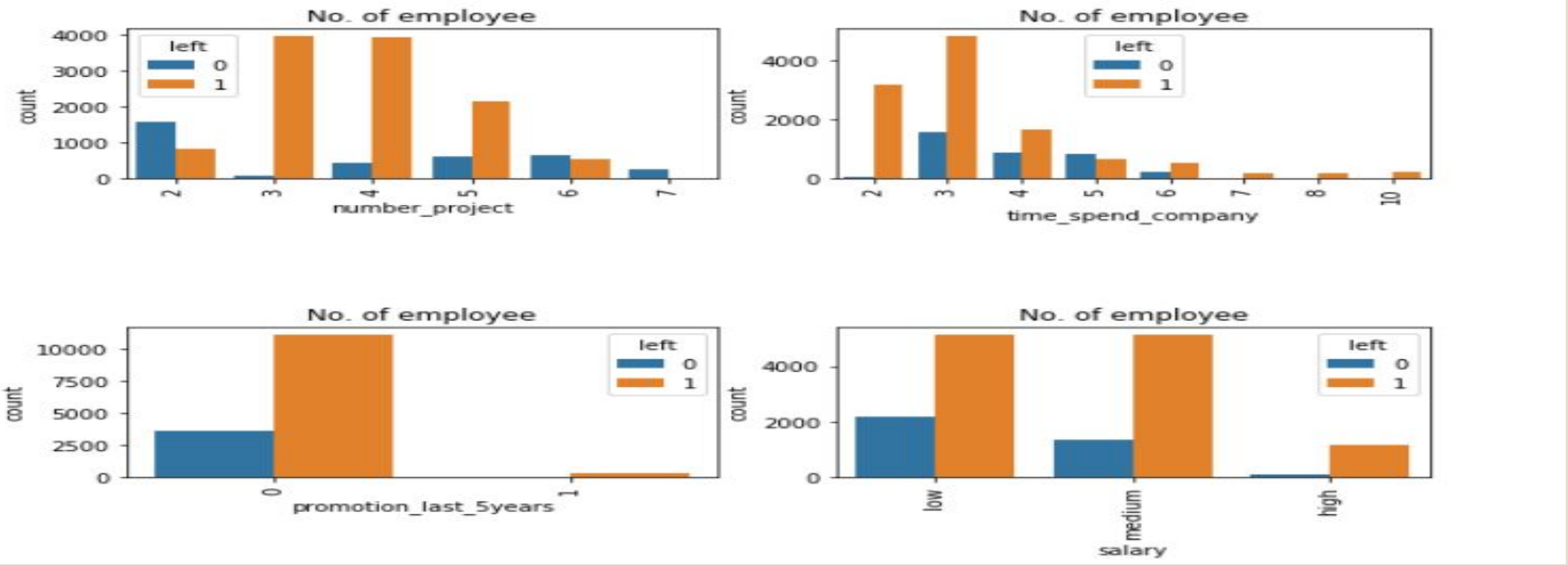
```
features=['promotion_last_5years','salary']
fig=plt.subplots(figsize=(10,15))
for i, j in enumerate(features):
    plt.subplot(4, 2, i+1)
    plt.subplots_adjust(hspace = 1.0)
    sns.countplot(x=j,data = data)
    plt.xticks(rotation=90)
    plt.title("No. of employee")
```



- We can see that the number of employees that have not been promoted is at a high rate. Another clue why the left.
- Lets check if these employee are paid well.
- most of these employee earn low or medium salary.

- Lets see the number of project and how they both handle and react to it.

```
features=['number_project','time_spend_company','promotion_last_5years','salary']
fig=plt.subplots(figsize=(10,15))
for i, j in enumerate(features):
    plt.subplot(4, 2, i+1)
    plt.subplots_adjust(hspace = 1.0)
    sns.countplot(x=j,data = data, hue='left')
    plt.xticks(rotation=90)
    plt.title("No. of employee")
```



- look at the data on the existing employee. We can see that they are only comfortable with 3-4 project, as soon as it gets to 5, they begin to leave, and when it gets to 6 project , they are all gone. They sure don't like to be overworked.
- We can see that after 5years without promotion they begin to leave.
- Most of them spent 2-4 years before they leave.

What we know so far

- The employee don't like to be over worked. Project range between 3-5 seems to be ok for them, the don't like to be over worked.
- we also noticed that they will quit if not promoted within 5 years.
- Most off the employee that left are low salary earners.
- We can also tell that the 3 year mark is very important in their life. That's the point when they decide, if they want to leave the company. So in that time we either increase their salary, promote them, or reduce their workload.
- So we can see that satisfaction and performance is part of the reason they left.

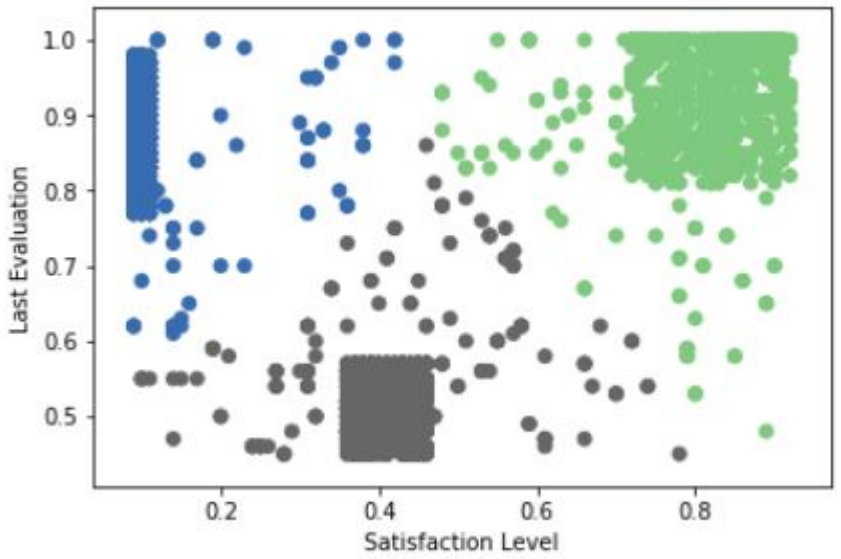
Cluster analysis

Lets group the employees that left into clusters and see the type of employee they are. We will use two variables, satisfaction level and last evaluation.

Am going to use tableau public and sklearn.cluster and import Kmeans algorithm to get the details in the employee that left.

```
#import module
from sklearn.cluster import KMeans
# Filter data
left_emp =  data[['satisfaction_level', 'last_evaluation']][data.left == 0]
# Create groups using K-means clustering.
kmeans = KMeans(n_clusters = 3, random_state = 0).fit(left_emp)

# Add new column "label" annd assign cluster labels.
left_emp['label'] = kmeans.labels_
# Draw scatter plot
plt.scatter(left_emp['satisfaction_level'], left_emp['last_evaluation'], c=left_emp['label'],cmap='Accent')
plt.xlabel('Satisfaction_Level')
plt.ylabel('Last_Evaluation')
```



- High Satisfaction and High Evaluation(Shaded in green color). These set of people got better offer from some where else, they are low salary earner with no promotion in the last 5 years. they wanted more. That's why they left.
- Low Satisfaction and High Evaluation(Shaded in blue color). These set of people are valued by the company but are not satisfied with the way they are treated. they handle 5-7 project, have spent 4-5years with the company and are paid low salary. they are overworked and they haven't been promoted in last five years, that's why they are dissatisfied.
- Moderate Satisfaction and moderate Evaluation (Shaded in grey color).these people are somewhere in between, they are not motivated by they job. They handle an average of 2 project and salary is low or medium. They want more responsibility.

Preparing data for building the model.

- So with what we know on those employees that left, I think we can build a model that can predict type of employee that is likely to leave the company.
- Before we build a model we have to organize the data. We have some attribute that has string values such as salary and department. We need them to build a model so we can not ignore them. We have to find a way to encode strings to numeric.
- Salary column's value can be represented as low:0, medium:1, and high:2.
- We can either do these manually or we can use label encoding which is a function of sklearn.
- So we then group the attributes in to “x” and “y”. “x” been the input and “y” been the outcome.
- **input**: will include satisfaction_level, last_evaluation, number_project, average_montly_hours, time_spend_company, Work_accident, promotion_last_5years, Departments , salary.
- **outcome**: will be the “left”.
- The we split the data into training and testing sets. 70:30 proportion. With 30% been the test data.

```
from sklearn import metrics
# Import LabelEncoder
from sklearn import preprocessing
#creating labelEncoder
labelE = preprocessing.LabelEncoder()
# Converting string labels into numbers.
data['salary']=labelE.fit_transform(data['salary'])
data['dept']=labelE.fit_transform(data['dept'])
#Spliting the Features data
X=data[['satisfaction_level', 'last_evaluation', 'number_project',
       'average_montly_hours', 'time_spend_company',
       'promotion_last_5years','salary']]
y=data['left']

# Import train test split function
from sklearn.model_selection import train_test_split

# Split dataset into training and test set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
from sklearn.ensemble import GradientBoostingClassifier
gb = GradientBoostingClassifier()
#Train the model using the training sets
gb.fit(X_train, y_train)
#Predict the response for test dataset
y_pred = gb.predict(X_test)
print("accuracy:",metrics.accuracy_score(y_test, y_pred))
print("precision:",metrics.precision_score(y_test, y_pred))
print("recall:",metrics.recall_score(y_test, y_pred))
```

MODEL OUTPUT

```
accuracy: 0.9742222222222222
precision: 0.9752660339373023
recall: 0.9912306343174511
```

Evaluating Model Performance

- We check the Accuracy, to see, how often the classifier is correct?

And the model's accuracy is 97%

We have to check for the precision, cause I want to know, how precise the model's prediction is?

Precision is at 95%,

Recall: the gradient booster can identity any employee who left in the test set 92% of the time.

cross checking these result with a decision tree algorithm just to be sure.

```
le = preprocessing.LabelEncoder()
 # Converting string labels into numbers.
data['salary']=le.fit_transform(data['salary'])
data['dept']=le.fit_transform(data['dept'])

input =data[['satisfaction_level', 'last_evaluation', 'number_project',
        'average_montly_hours', 'time_spend_company', 'Work_accident',
        'promotion_last_5years', 'dept', 'salary']]
outcome =data['left']
# spliting my data into training and testing data training=70%, testing=30%
(input_train, input_test, outcome_train, outcome_test) = train_test_split(input,outcome, test_size=0.3)
# building the model
model = DecisionTreeClassifier()
fitmodel = model.fit(input_train, outcome_train)
predictions = fitmodel.predict(input_test)
# check for the percentage of accuracy of model
print(accuracy_score(outcome_test, predictions))

from sklearn import metrics

print("Precision:",metrics.precision_score(outcome_test, predictions))
# Model Recall
print("Recall:",metrics.recall_score(outcome_test, predictions))
```

```
0.9784444444444444
Precision: 0.9886627906976744
Recall: 0.9832321480196589
```

our model seems to be working just fine.

CONCLUSION

What type of employees are leaving:

- employees that are not promoted within 5 years and have high evaluation, have spent at least 4-5years.
- Employee that are engaged in 5-7 project tends and are paid low income,
- Employee that are not handling 1-2 project don't feel engaged enough.

Employee that are prone to leave:

- The employee that are doing 5-6 project are going to leave,
- The ones that have not been promoted in within 5 year and have high evaluation will leave.
- Employee with high evaluation and low salary will leave.
- Employee with low project(1-2) with definitely leave.