

Introduction:

This project aims to analyze employee attrition within an organization using HR data. The purpose is to identify patterns and contributing factors behind attrition and suggest strategies to mitigate it. The insights generated help the HR team make data-driven decisions.

Abstract:

Employee attrition is a critical issue faced by organizations. This project leverages data visualization and statistical analysis to highlight key trends and roles most impacted by attrition. Based on these insights, actionable recommendations are provided to help reduce attrition rates and enhance employee retention.

Tools Used:

Microsoft Excel: for data cleaning and pivot analysis - Tableau: for interactive dashboards and visualizations - Python: for data manipulation and model

Steps Involved in Building the Project:

- Collected and cleaned HR employee attrition dataset
- Explored data using Excel to identify high-level trends
- Created dashboards and charts in Tableau for visualization
- Performed deeper analysis of attrition by job role and income
- Drafted actionable insights and prevention suggestions
- Compiled results and generated a final report

Conclusion:

Understanding and preventing employee attrition is essential for business sustainability. By leveraging analytics tools, this project provides HR departments with insights and targeted strategies to retain talent, reduce hiring costs, and build a more engaged workforce.

```
[1]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
[4]: df=pd.read_csv('/content/WA_Fn-UseC_-HR-Employee-Attrition.csv')
```

```
[5]: df.head(10)
```

```
[5]:
```

	Age	Attrition	BusinessTravel	DailyRate	Department	\
0	41	Yes	Travel_Rarely	1102		Sales
1	49	No	Travel_Frequently	279	Research & Development	
2	37	Yes	Travel_Rarely	1373	Research & Development	
3	33	No	Travel_Frequently	1392	Research & Development	
4	27	No	Travel_Rarely	591	Research & Development	
5	32	No	Travel_Frequently	1005	Research & Development	
6	59	No	Travel_Rarely	1324	Research & Development	
7	30	No	Travel_Rarely	1358	Research & Development	
8	38	No	Travel_Frequently	216	Research & Development	
9	36	No	Travel_Rarely	1299	Research & Development	

	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber	\
0		1	2 Life Sciences	1		1
1		8	1 Life Sciences	1		2
2		2	2 Other	1		4
3		3	4 Life Sciences	1		5
4		2	1 Medical	1		7
5		2	2 Life Sciences	1		8
6		3	3 Medical	1		10
7		24	1 Life Sciences	1		11
8		23	3 Life Sciences	1		12
9		27	3 Medical	1		13

	...	RelationshipSatisfaction	StandardHours	StockOptionLevel	\
0	...		1	80	0
1	...		4	80	1
2	...		2	80	0
3	...		3	80	0

4	...	4	80	1
5	...	3	80	0
6	...	1	80	3
7	...	2	80	1
8	...	2	80	0
9	...	2	80	2

	TotalWorkingYears	TrainingTimesLastYear	WorkLifeBalance	YearsAtCompany \
0	8	0	1	6
1	10	3	3	10
2	7	3	3	0
3	8	3	3	8
4	6	3	3	2
5	8	2	2	7
6	12	3	2	1
7	1	2	3	1
8	10	2	3	9
9	17	3	2	7

	YearsInCurrentRole	YearsSinceLastPromotion	YearsWithCurrManager
0	4	0	5
1	7	1	7
2	0	0	0
3	7	3	0
4	2	2	2
5	7	3	6
6	0	0	0
7	0	0	0
8	7	1	8
9	7	7	7

[10 rows x 35 columns]

```
[6]: df.shape
```

```
[6]: (1470, 35)
```

```
[7]: df.index
```

```
[7]: RangeIndex(start=0, stop=1470, step=1)
```

```
[8]: df.columns
```

```
[8]: Index(['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'],
dtype='object')

```

```
[9]: df.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1470 entries, 0 to 1469
Data columns (total 35 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Age                                   1470 non-null   int64
1   Attrition                           1470 non-null   object
2   BusinessTravel                       1470 non-null   object
3   DailyRate                           1470 non-null   int64
4   Department                          1470 non-null   object
5   DistanceFromHome                    1470 non-null   int64
6   Education                           1470 non-null   int64
7   EducationField                       1470 non-null   object
8   EmployeeCount                       1470 non-null   int64
9   EmployeeNumber                      1470 non-null   int64
10  EnvironmentSatisfaction              1470 non-null   int64
11  Gender                              1470 non-null   object
12  HourlyRate                          1470 non-null   int64
13  JobInvolvement                      1470 non-null   int64
14  JobLevel                            1470 non-null   int64
15  JobRole                             1470 non-null   object
16  JobSatisfaction                     1470 non-null   int64
17  MaritalStatus                       1470 non-null   object
18  MonthlyIncome                       1470 non-null   int64
19  MonthlyRate                         1470 non-null   int64
20  NumCompaniesWorked                  1470 non-null   int64
21  Over18                              1470 non-null   object
22  OverTime                            1470 non-null   object
23  PercentSalaryHike                   1470 non-null   int64
24  PerformanceRating                   1470 non-null   int64
25  RelationshipSatisfaction             1470 non-null   int64
26  StandardHours                       1470 non-null   int64
27  StockOptionLevel                    1470 non-null   int64
28  TotalWorkingYears                   1470 non-null   int64
29  TrainingTimesLastYear               1470 non-null   int64
30  WorkLifeBalance                     1470 non-null   int64
31  YearsAtCompany                      1470 non-null   int64

```

```

32 YearsInCurrentRole      1470 non-null   int64
33 YearsSinceLastPromotion  1470 non-null   int64
34 YearsWithCurrManager    1470 non-null   int64
dtypes: int64(26), object(9)
memory usage: 402.1+ KB

```

```
[10]: df.describe()
```

```

[10]:
count      Age      DailyRate  DistanceFromHome  Education  EmployeeCount  \
count  1470.000000  1470.000000      1470.000000  1470.000000      1470.0
mean    36.923810   802.485714        9.192517    2.912925        1.0
std      9.135373   403.509100        8.106864    1.024165        0.0
min     18.000000   102.000000        1.000000    1.000000        1.0
25%     30.000000   465.000000        2.000000    2.000000        1.0
50%     36.000000   802.000000        7.000000    3.000000        1.0
75%     43.000000  1157.000000       14.000000    4.000000        1.0
max     60.000000  1499.000000       29.000000    5.000000        1.0

```

```

count      EmployeeNumber  EnvironmentSatisfaction  HourlyRate  JobInvolvement  \
count  1470.000000      1470.000000  1470.000000  1470.000000
mean    1024.865306        2.721769    65.891156    2.729932
std      602.024335        1.093082    20.329428    0.711561
min         1.000000        1.000000    30.000000    1.000000
25%      491.250000        2.000000    48.000000    2.000000
50%     1020.500000        3.000000    66.000000    3.000000
75%     1555.750000        4.000000    83.750000    3.000000
max     2068.000000        4.000000   100.000000    4.000000

```

```

count      JobLevel  ...  RelationshipSatisfaction  StandardHours  \
count  1470.000000  ...      1470.000000      1470.0
mean     2.063946  ...        2.712245        80.0
std      1.106940  ...        1.081209         0.0
min      1.000000  ...        1.000000        80.0
25%      1.000000  ...        2.000000        80.0
50%      2.000000  ...        3.000000        80.0
75%      3.000000  ...        4.000000        80.0
max      5.000000  ...        4.000000        80.0

```

```

count      StockOptionLevel  TotalWorkingYears  TrainingTimesLastYear  \
count  1470.000000      1470.000000      1470.000000
mean      0.793878        11.279592        2.799320
std      0.852077         7.780782        1.289271
min      0.000000         0.000000        0.000000
25%      0.000000         6.000000        2.000000
50%      1.000000        10.000000        3.000000
75%      1.000000        15.000000        3.000000
max      3.000000        40.000000        6.000000

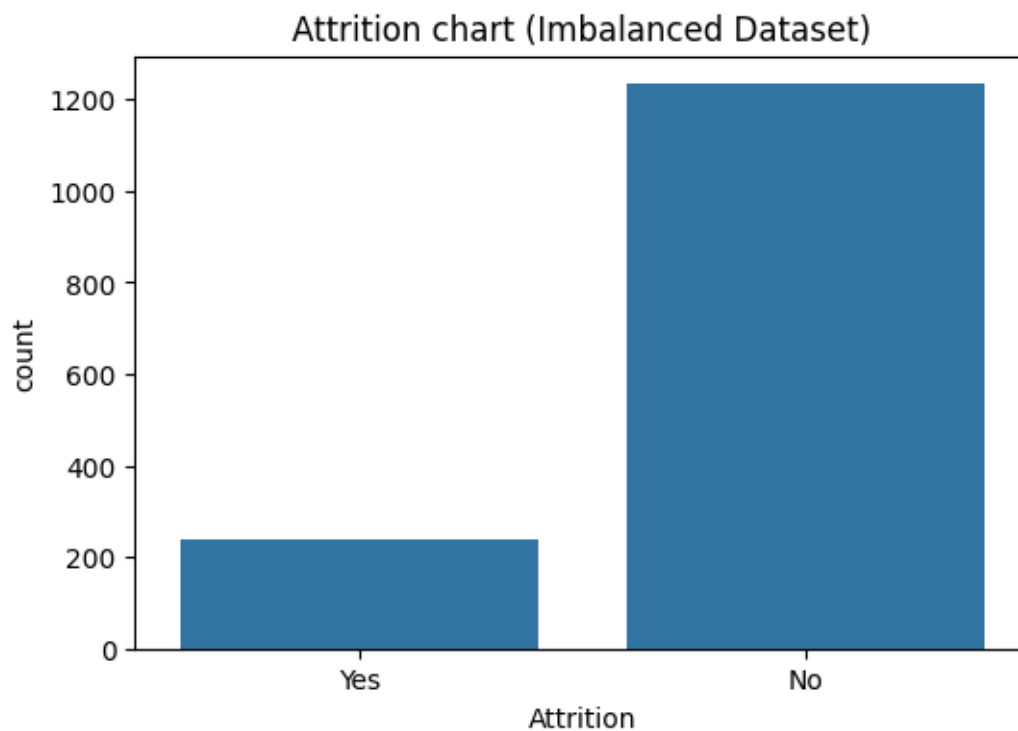
```

	WorkLifeBalance	YearsAtCompany	YearsInCurrentRole \
count	1470.000000	1470.000000	1470.000000
mean	2.761224	7.008163	4.229252
std	0.706476	6.126525	3.623137
min	1.000000	0.000000	0.000000
25%	2.000000	3.000000	2.000000
50%	3.000000	5.000000	3.000000
75%	3.000000	9.000000	7.000000
max	4.000000	40.000000	18.000000

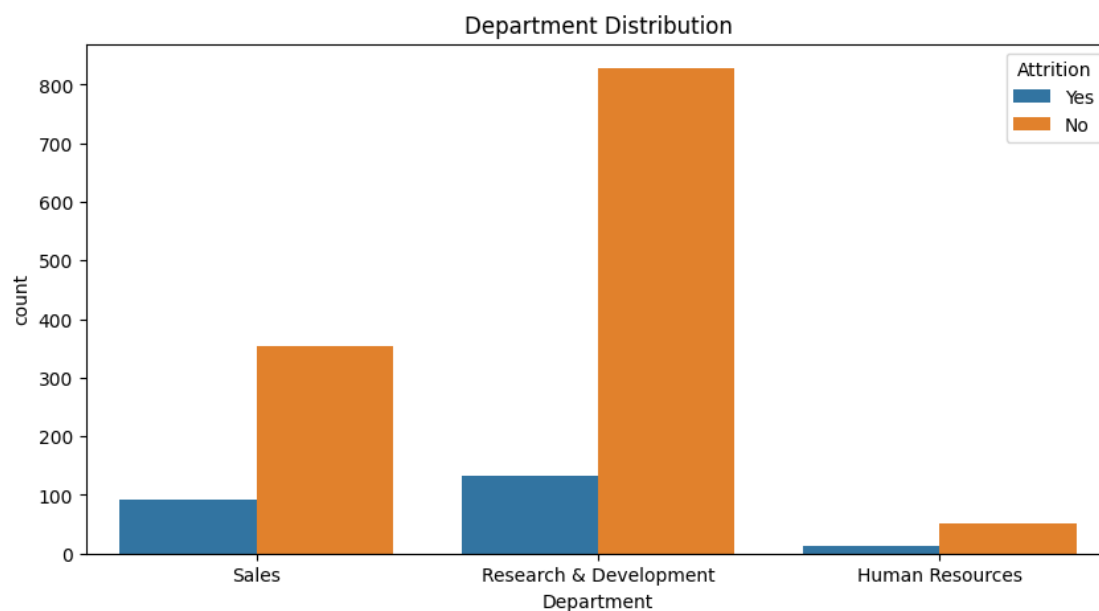
	YearsSinceLastPromotion	YearsWithCurrManager
count	1470.000000	1470.000000
mean	2.187755	4.123129
std	3.222430	3.568136
min	0.000000	0.000000
25%	0.000000	2.000000
50%	1.000000	3.000000
75%	3.000000	7.000000
max	15.000000	17.000000

[8 rows x 26 columns]

```
[11]: # Attrition distribution
plt.figure(figsize=(6, 4))
sns.countplot(data=df, x='Attrition')
plt.title("Attrition chart (Imbalanced Dataset)")
plt.show()
```



```
[12]: plt.figure(figsize=(10,5))
sns.countplot(data=df, x='Department', hue='Attrition')
plt.title("Department Distribution")
plt.show()
```




```
[13]: # Convert 'Attrition' to numeric for correlation
df['AttritionNumeric'] = df['Attrition'].map({'Yes': 1, 'No': 0})

# Correlation of all numeric features with Attrition
corr_matrix = df.corr(numeric_only=True)['AttritionNumeric'].
    ↪sort_values(ascending=False)

# Display with gradient
corr_matrix.to_frame().style.background_gradient(cmap='coolwarm')
```

```
[13]: <pandas.io.formats.style.Styler at 0x786e38340950>
```

```
[15]: df.isnull().sum()
```

```
[15]: Age                                0
Attrition                              0
BusinessTravel                         0
DailyRate                             0
Department                             0
DistanceFromHome                       0
Education                              0
EducationField                         0
EmployeeCount                          0
EmployeeNumber                         0
EnvironmentSatisfaction                0
Gender                                  0
HourlyRate                             0
JobInvolvement                         0
JobLevel                               0
JobRole                                0
JobSatisfaction                        0
MaritalStatus                         0
MonthlyIncome                         0
MonthlyRate                           0
NumCompaniesWorked                    0
Over18                                 0
OverTime                               0
PercentSalaryHike                     0
PerformanceRating                     0
RelationshipSatisfaction               0
StandardHours                         0
StockOptionLevel                      0
TotalWorkingYears                     0
TrainingTimesLastYear                 0
WorkLifeBalance                       0
YearsAtCompany                        0
```

```

YearsInCurrentRole      0
YearsSinceLastPromotion  0
YearsWithCurrManager    0
AttritionNumeric        0
SalaryBand              0
dtype: int64

```

```

[14]: # Create salary bands for analysis
df['SalaryBand'] = pd.qcut(df['MonthlyIncome'], q=4, labels=['Low', 'Medium', 'High', 'Very High'])

# 1. Department-wise Attrition
dept_attrition = df.groupby(['Department', 'Attrition']).size().unstack(fill_value=0)
dept_attrition_percent = dept_attrition.div(dept_attrition.sum(axis=1), axis=0) * 100
dept_attrition_percent

```

```

[14]: Attrition      No      Yes
Department
Human Resources    80.952381  19.047619
Research & Development  86.160250  13.839750
Sales              79.372197  20.627803

```

```

[16]: # 2. Salary Band vs Attrition
salary_attrition = df.groupby(['SalaryBand', 'Attrition']).size().unstack(fill_value=0)
salary_attrition_percent = salary_attrition.div(salary_attrition.sum(axis=1), axis=0) * 100

# 3. Promotion vs Attrition
# Consider "YearsSinceLastPromotion" == 0 as promoted recently
df['RecentlyPromoted'] = df['YearsSinceLastPromotion'] == 0
promotion_attrition = df.groupby(['RecentlyPromoted', 'Attrition']).size().unstack(fill_value=0)
promotion_attrition_percent = promotion_attrition.div(promotion_attrition.sum(axis=1), axis=0) * 100

```

<ipython-input-16-0824106cd8fd>:2: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.

```

salary_attrition = df.groupby(['SalaryBand', 'Attrition']).size().unstack(fill_value=0)

```

```

[17]: salary_attrition_percent, promotion_attrition_percent

```

```

[17]: (Attrition      No      Yes
SalaryBand
Low      70.731707  29.268293
Medium   85.792350  14.207650
High     89.373297  10.626703
Very High 89.673913  10.326087,
Attrition      No      Yes
RecentlyPromoted
False          85.714286  14.285714
True           81.067126  18.932874)

```

```

[17]: 

```

HR Attrition Analysis & Prevention Suggestions

1. Key Attrition Insights by Job Role

- Sales Executive and Laboratory Technician roles show highest attrition rates.
- Research Scientist and Healthcare Representative also show notable attrition.
- Roles like Research Director, Manager, and Manufacturing Director have low attrition.

2. Income Patterns

- Some roles with high attrition (e.g., Sales Executive, Laboratory Technician) do not have the highest income.
- Compensation dissatisfaction may be a contributing factor.
- Sales Executive has the highest headcount, implying its attrition has a large organizational impact.

3. Suggestions to Reduce Attrition

Job Satisfaction & Engagement

- Conduct regular pulse surveys in high-attrition departments.
- Offer growth paths and skill development programs for Lab Technicians & Sales Executives.

Compensation Strategy

- Benchmark industry salaries for high-risk roles and adjust incentives/bonuses.
- Provide transparent performance-linked raises.

Work-Life Balance

- Promote flexible work hours or hybrid work models where possible.
- Offer mental health and wellness support programs.

Career Development

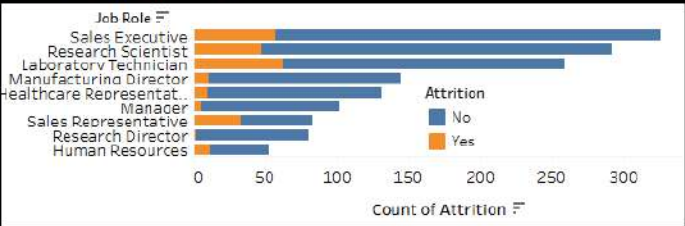
- Launch internal career mobility programs for employees feeling stuck in current roles.
- Encourage cross-training and certifications (especially for Research and Healthcare roles).

Managerial Coaching

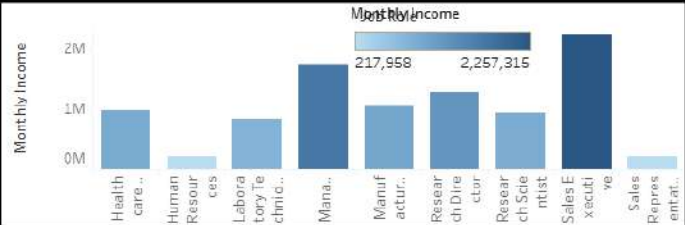
- Improve leadership and communication training for managers of high-attrition departments.

HR Analytics

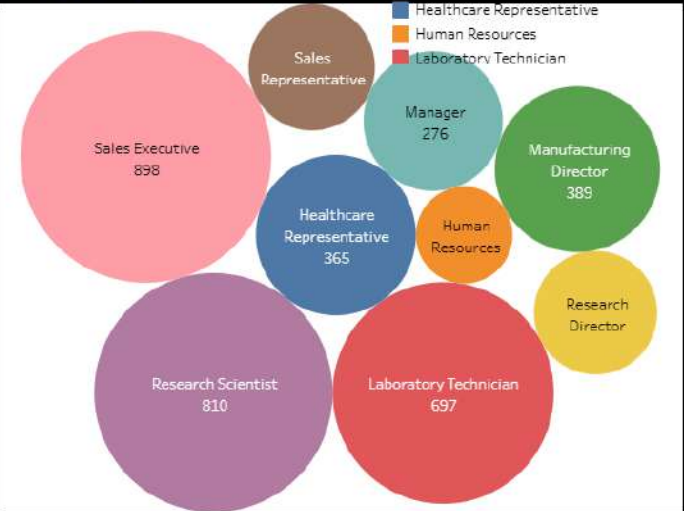
Bar chart(horizontal)



Bar graph



Bubble chart



Highlight Job Role

Highlight Job Role