**Unified Mentor**

Green Destination Attrition Data Analysis

By Tahir Malik

# Problem Statement

Green Destinations is a well-known travel agency. The HR Director has recently noticed and increase in employees leaving (attrition)

She would like to figure out any trends or patterns. She has surveyed the staff of Green Destinations and provided you with the data. She would like to know what the attrition rate is (% of people who left). She would also like to know if factors like age, years at the company and income play a part in determining if people will leave or not.

# Python Code

*#importing python libraries*

import pandas as pd

*#importing dataset into dataframe variable*

dataframe = pd.read\_csv('./csv/greendestination-dataset.csv')

*#printing top 5 records from the top of dataset*

dataframe.head(5)

*#removing useless columns*

del dataframe['EmployeeCount']

del dataframe['StandardHours']

del dataframe['Over18']

dataframe.head(5)

*#printing rows & columns of our dataset*

print(f"Rows: {dataframe.shape[0]}\nColumns: {dataframe.shape[1]}")

*#checking whether dataset has any NULL values*

dataframe.isnull().sum()

dataframe.head(5)

*#finding number of employee's*

print(f"No. of employee's: {len(dataframe)}")

*#finding number of attrition's*

attrition\_count = 0

for i in range(len(dataframe['Attrition'])):

    if dataframe['Attrition'][i] == 'Yes':

        attrition\_count += 1

print(f"No. of attrition: {attrition\_count}")

*#finding attrition rate*

print(f"Attrition rate: {round((attrition\_count/len(dataframe['Attrition']))\*100, 1)} %")

*#finding average age*

print(f"Average age: {round(dataframe['Age'].mean(), 0).astype(int)}")

*#finding average salary*

print(f"Average salary: $ {round(dataframe['MonthlyIncome'].mean(), 0).astype(int)}")

*#finding average years*

print(f"Average years: {round(dataframe['YearsAtCompany'].mean(), 1)}")

*#segregating employees by gender*

gender\_count = dataframe['Gender'].value\_counts()

print(f"Male: {gender\_count.iloc[0]}")

print(f"Female: {gender\_count.iloc[1]}")

*#finding attrition by gender*

male\_attrition\_count = 0

female\_attrition\_count = 0

for i in range(len(dataframe)):

    if dataframe['Attrition'][i] == 'Yes':

        if dataframe['Gender'][i] == 'Male':

            male\_attrition\_count += 1

        elif dataframe['Gender'][i] == 'Female':

            female\_attrition\_count += 1

print("Attrition by Gender:")

print(f"Male: {male\_attrition\_count}")

print(f"Female: {female\_attrition\_count}")

*#finding attrition by job role*

unique\_job\_role = dataframe['JobRole'].unique()

print(f"No. of job roles: {len(unique\_job\_role)}")

print()

sales\_executive\_count = 0

research\_scientist\_count = 0

laboratory\_technician\_count = 0

manufacturing\_director\_count = 0

healthcare\_representative\_count = 0

manager\_count = 0

sales\_representative\_count = 0

research\_director\_count = 0

human\_resources\_count = 0

for i in range(len(dataframe)):

    if dataframe['Attrition'][i] == 'Yes':

        if dataframe['JobRole'][i] == 'Sales Executive':

            sales\_executive\_count += 1

        elif dataframe['JobRole'][i] == 'Research Scientist':

            research\_scientist\_count += 1

        elif dataframe['JobRole'][i] == 'Laboratory Technician':

            laboratory\_technician\_count += 1

        elif dataframe['JobRole'][i] == 'Manufacturing Director':

            manufacturing\_director\_count += 1

        elif dataframe['JobRole'][i] == 'Healthcare Representative':

            healthcare\_representative\_count += 1

        elif dataframe['JobRole'][i] == 'Manager':

            manager\_count += 1

        elif dataframe['JobRole'][i] == 'Sales Representative':

            sales\_representative\_count += 1

        elif dataframe['JobRole'][i] == 'Research Director':

            research\_director\_count += 1

        elif dataframe['JobRole'][i] == 'Human Resources':

            human\_resources\_count += 1

print("Attrition by Job Role:")

print(f"Sales Executive: {sales\_executive\_count}")

print(f"Research Scientist: {research\_scientist\_count}")

print(f"Laboratory Technician: {laboratory\_technician\_count}")

print(f"Manufacturing Director: {manufacturing\_director\_count}")

print(f"Healthcare Representative: {healthcare\_representative\_count}")

print(f"Manager: {manager\_count}")

print(f"Sales Representative: {sales\_representative\_count}")

print(f"Research Director: {research\_director\_count}")

print(f"Human Resources: {human\_resources\_count}")

*#finding attrition by age*

count\_age\_18\_to\_25 = 0

count\_age\_26\_to\_35 = 0

count\_age\_36\_to\_45 = 0

count\_age\_46\_to\_55 = 0

count\_age\_56\_to\_60 = 0

for i in range(len(dataframe)):

    if dataframe['Attrition'][i] == 'Yes':

        if dataframe['Age'][i] >= 18 and dataframe['Age'][i] <= 25:

            count\_age\_18\_to\_25 += 1

        elif dataframe['Age'][i] >= 26 and dataframe['Age'][i] <= 35:

            count\_age\_26\_to\_35 += 1

        elif dataframe['Age'][i] >= 36 and dataframe['Age'][i] <= 45:

            count\_age\_36\_to\_45 += 1

        elif dataframe['Age'][i] >= 46 and dataframe['Age'][i] <= 55:

            count\_age\_46\_to\_55 += 1

        elif dataframe['Age'][i] >= 56 and dataframe['Age'][i] <= 60:

            count\_age\_56\_to\_60 += 1

print(f"Attrition from (18 - 25) yrs: {count\_age\_18\_to\_25}")

print(f"Attrition from (26 - 35) yrs: {count\_age\_26\_to\_35}")

print(f"Attrition from (36 - 45) yrs: {count\_age\_36\_to\_45}")

print(f"Attrition from (46 - 55) yrs: {count\_age\_46\_to\_55}")

print(f"Attrition from (56 - 60) yrs: {count\_age\_56\_to\_60}")

*#finding attrition by salary*

count\_monthly\_income\_upto\_2k = 0

count\_monthly\_income\_upto\_5k = 0

count\_monthly\_income\_upto\_10k = 0

count\_monthly\_income\_upto\_15k = 0

count\_monthly\_income\_upto\_15kplus = 0

for i in range(len(dataframe)):

    if dataframe['Attrition'][i] == 'Yes':

        if dataframe['MonthlyIncome'][i] <= 2000:

            count\_monthly\_income\_upto\_2k += 1

        elif dataframe['MonthlyIncome'][i] >= 2001 and dataframe['MonthlyIncome'][i] <= 5000:

            count\_monthly\_income\_upto\_5k += 1

        elif dataframe['MonthlyIncome'][i] >= 5001 and dataframe['MonthlyIncome'][i] <= 10000:

            count\_monthly\_income\_upto\_10k += 1

        elif dataframe['MonthlyIncome'][i] >= 10001 and dataframe['MonthlyIncome'][i] <= 15000:

            count\_monthly\_income\_upto\_15k += 1

        elif dataframe['MonthlyIncome'][i] >= 15001:

            count\_monthly\_income\_upto\_15kplus += 1

print(f"Attrition salary band from upto 2k: {count\_monthly\_income\_upto\_2k}")

print(f"Attrition salary band from 2k - 5k: {count\_monthly\_income\_upto\_5k}")

print(f"Attrition salary band from 5k - 10k: {count\_monthly\_income\_upto\_10k}")

print(f"Attrition salary band from 10k - 15k: {count\_monthly\_income\_upto\_15k}")

print(f"Attrition salary band from 15k plus: {count\_monthly\_income\_upto\_15kplus}")

*#finding attrition by education*

unique\_education = dataframe['EducationField'].unique()

print(f"No. of education field: {len(unique\_education)}")

print()

life\_sciences\_edu\_count = 0

others\_edu\_count = 0

medical\_edu\_count = 0

marketing\_edu\_count = 0

technical\_degree\_edu\_count = 0

human\_resources\_edu\_count = 0

for i in range(len(dataframe)):

    if dataframe['Attrition'][i] == 'Yes':

        if dataframe['EducationField'][i] == 'Life Sciences':

            life\_sciences\_edu\_count += 1

        elif dataframe['EducationField'][i] == 'Other':

            others\_edu\_count += 1

        elif dataframe['EducationField'][i] == 'Medical':

            medical\_edu\_count += 1

        elif dataframe['EducationField'][i] == 'Marketing':

            marketing\_edu\_count += 1

        elif dataframe['EducationField'][i] == 'Technical Degree':

            technical\_degree\_edu\_count += 1

        elif dataframe['EducationField'][i] == 'Human Resources':

            human\_resources\_edu\_count += 1

print("Attrition by Education:")

print(f"Life Sciences: {life\_sciences\_edu\_count}")

print(f"Others: {others\_edu\_count}")

print(f"Medical: {medical\_edu\_count}")

print(f"Marketing: {marketing\_edu\_count}")

print(f"Technical Degree: {technical\_degree\_edu\_count}")

print(f"Human Resources: {human\_resources\_edu\_count}")

dataframe.to\_csv('./csv/cleaned-greendestination-dataset.csv', index = False)

**Insights**

- No. of employees: 1470

- No. of attrition: 237

- Attrition rate: 16.1 %

- Average age: 37

- Average salary: $ 6503

- Average years: 7.0

- Male: 882, Female: 588

- Attrition by Gender: Male: 150, Female: 87

- No. of job roles: 9

- Attrition by Job Role:

  1. Sales Executive: 57

  2. Research Scientist: 47

  3. Laboratory Technician: 62

  4. Manufacturing Director: 10

  5. Healthcare Representative: 9

  6. Manager: 5

  7. Sales Representative: 33

  8. Research Director: 2

  9. Human Resources: 12

- Attrition by Age:

  1. (18 - 25) yrs: 44

  2. (26 - 35) yrs: 116

  3. (36 - 45) yrs: 43

  4. (46 - 55) yrs: 26

  5. (56 - 60) yrs: 8

- Attrition by Salary:

  1. upto 2k: 18

  2. 2k - 5k: 145

  3. 5k - 10k: 49

  4. 10k - 15k: 20

  5. 15k plus: 5

- No. of education field: 6

- Attrition by Education:

  1. Life Sciences: 89

  2. Others: 11

  3. Medical: 63

  4. Marketing: 35

  5. Technical Degree: 32

  6. Human Resources: 7

**Conclusion**

Based on these insights, it appears that factors such as age, gender, salary, job role, and education field significantly influence attrition rates within the organization. Further analysis could delve into reasons behind these trends and formulate strategies to mitigate attrition, such as targeted retention programs, career development initiatives, and salary adjustments.