In [317…
```
import os
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import missingno as msno
```

In [318…
```
df = pd.read_csv(r"C:\Users\user\OneDrive\Desktop\UM Final Project\green destina
```

In [283…
```
df.head()
```

Out[283…

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Educat
0	41	Yes	Travel_Rarely	1102	Sales	1	
1	49	No	Travel_Frequently	279	Research & Development	8	
2	37	Yes	Travel_Rarely	1373	Research & Development	2	
3	33	No	Travel_Frequently	1392	Research & Development	3	
4	27	No	Travel_Rarely	591	Research & Development	2	

5 rows × 35 columns

In [319…
```
columns_to_drop = ['EmployeeCount', 'EmployeeNumber','StandardHours','Over18']
data = df.drop(columns=columns_to_drop)
```

In [338…
```
data.describe()
```

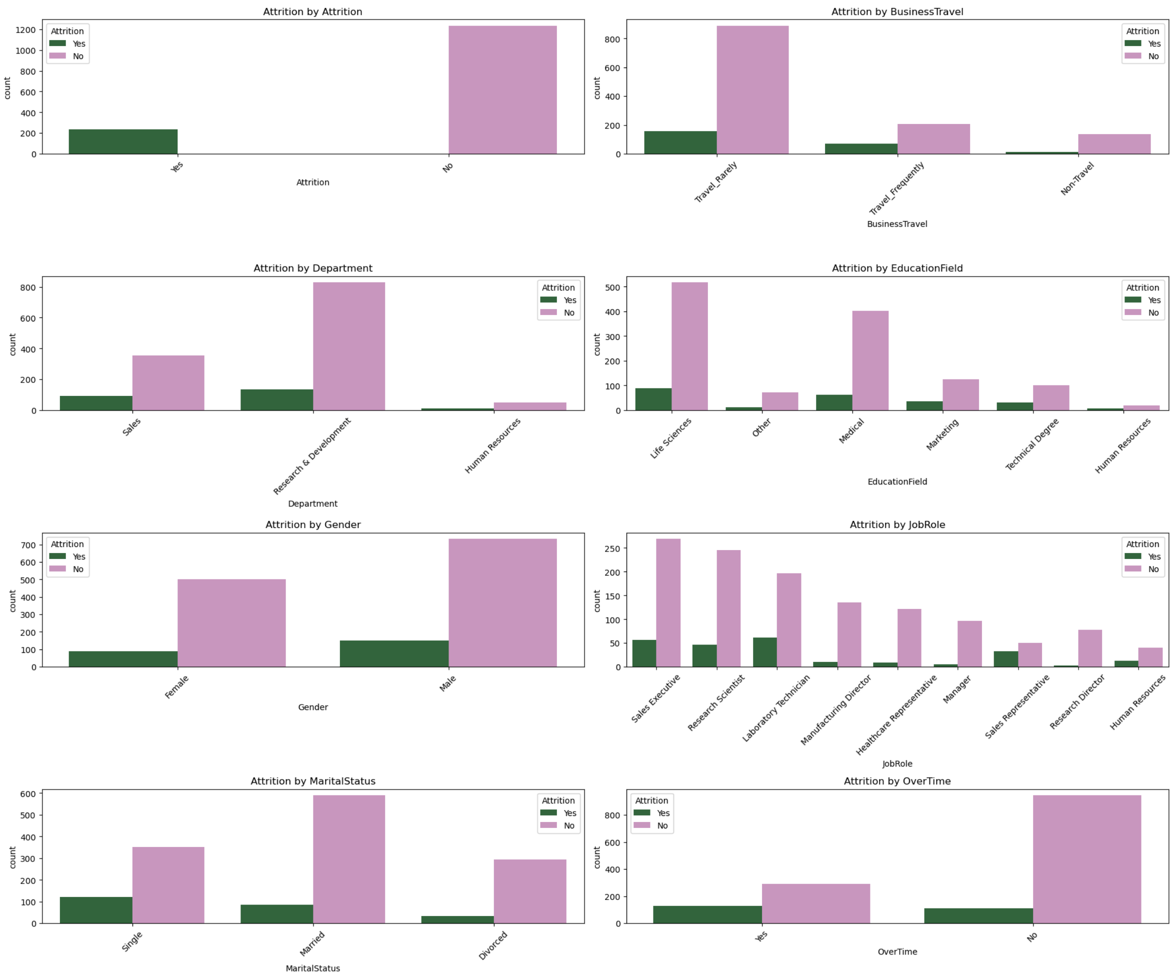
Out[338…

	Age	DailyRate	DistanceFromHome	Education	EnvironmentSatisfact
count	1470.000000	1470.000000	1470.000000	1470.000000	1470.000
mean	36.923810	802.485714	9.192517	2.912925	2.721
std	9.135373	403.509100	8.106864	1.024165	1.093
min	18.000000	102.000000	1.000000	1.000000	1.000
25%	30.000000	465.000000	2.000000	2.000000	2.000
50%	36.000000	802.000000	7.000000	3.000000	3.000
75%	43.000000	1157.000000	14.000000	4.000000	4.000
max	60.000000	1499.000000	29.000000	5.000000	4.000

8 rows × 23 columns

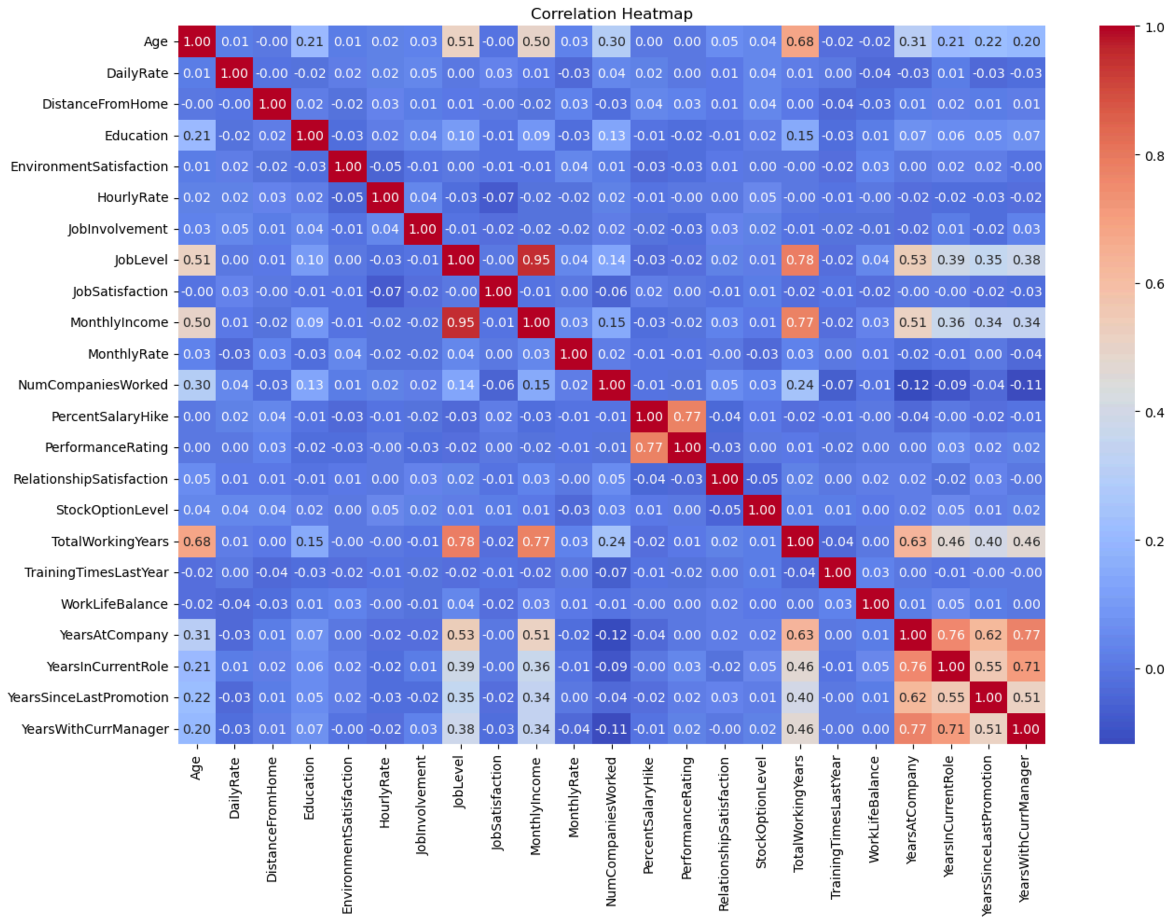
In [285…

```
categorical_cols = data.select_dtypes(include=['object']).columns
numerical_cols = data.select_dtypes(include=['int64', 'float64']).columns

# Categorical columns for visualization
plt.figure(figsize=(20, 20))
for i, col in enumerate(categorical_cols, 1):
    plt.subplot(len(categorical_cols) // 2 + 1, 2, i)
    sns.countplot(data=data, x=col, hue='Attrition', palette="cubehelix")
    plt.title(f'Attrition by {col}')
    plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



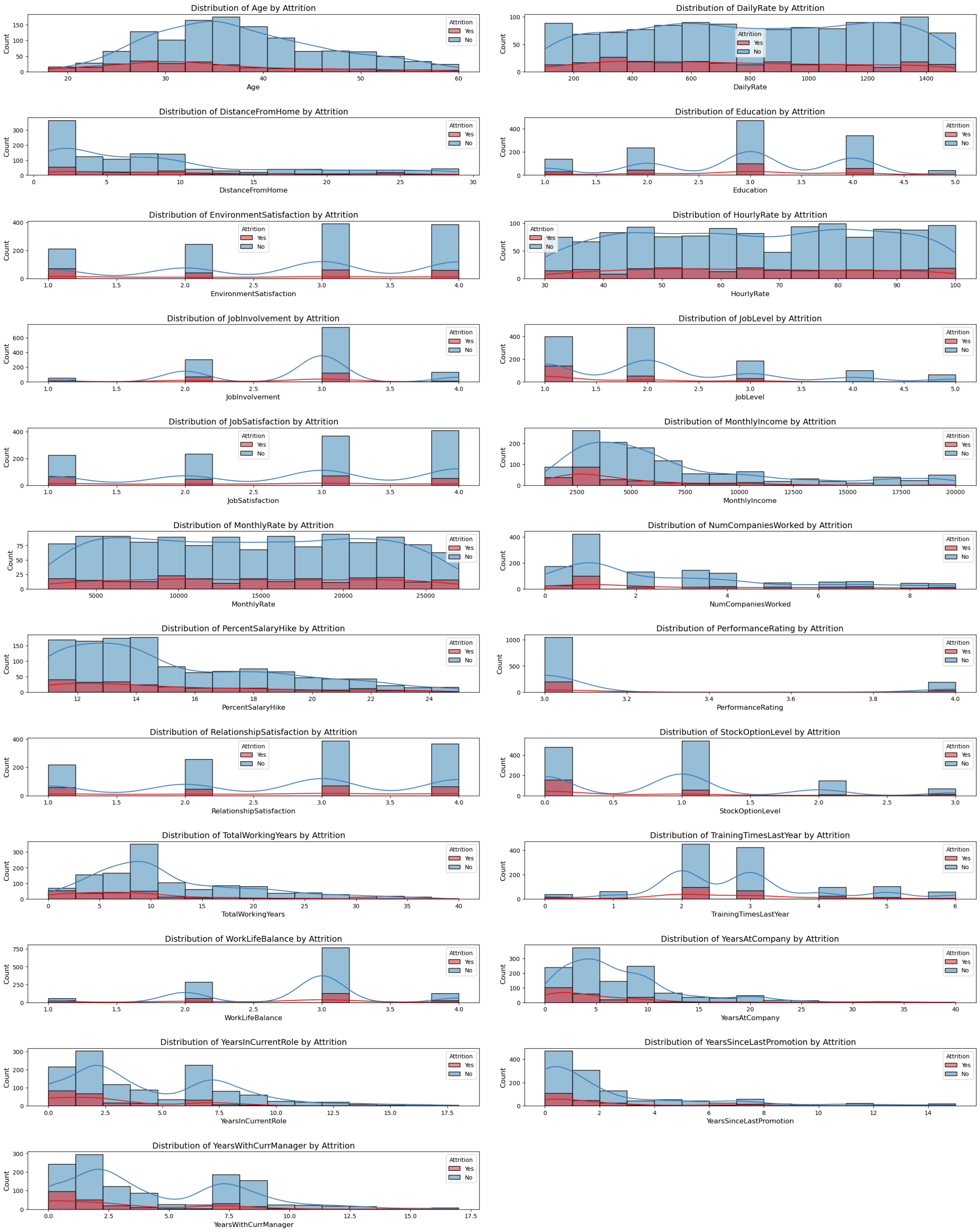
In [286…

```
plt.figure(figsize=(15, 10))
corr = data[numerical_cols].corr()
sns.heatmap(corr, annot=True, cmap="coolwarm", fmt=".2f")
plt.title("Correlation Heatmap")
plt.show()
```



In [287…

```python
# Numeric columns for visualization
plt.figure(figsize=(24, 30))
for i, col in enumerate(numerical_cols, 1):
    plt.subplot(len(numerical_cols) // 2 + 1, 2, i)
    sns.histplot(data=data, x=col, hue='Attrition', kde=True, bins=15, palette="
    plt.title(f'Distribution of {col} by Attrition', fontsize=14)
    plt.xlabel(col, fontsize=12)
    plt.ylabel('Count', fontsize=12)
    plt.xticks(fontsize=10)
    plt.yticks(fontsize=10)
plt.tight_layout(pad=3.0)
plt.show()
```



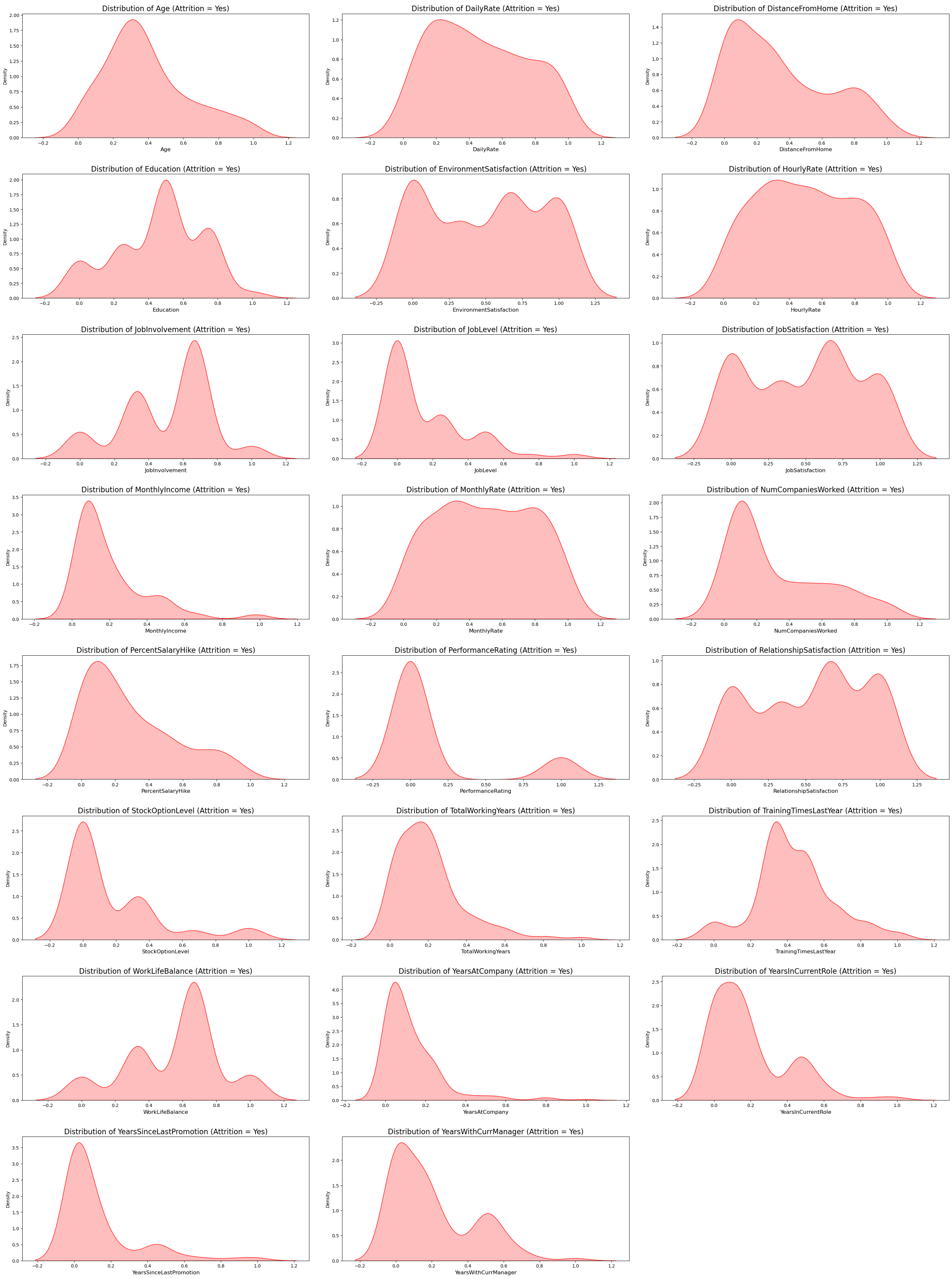
In [31]:

```
"""
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()
data_yes[numerical_cols] = scaler.fit_transform(data_yes[numerical_cols])
"""
```

```
C:\Users\user\AppData\Local\Temp\ipykernel_18684\796893097.py:4: SettingWithCopyW
arning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stabl
e/user_guide/indexing.html#returning-a-view-versus-a-copy
  data_yes[numerical_cols] = scaler.fit_transform(data_yes[numerical_cols])
```

In [288…

```python
plt.figure(figsize=(30, 40))
for i, col in enumerate(numerical_cols, 1):
    plt.subplot(len(numerical_cols) // 3 + 1, 3, i)
    sns.kdeplot(data=data_yes, x=col, fill=True, color="red")
    plt.title(f'Distribution of {col} (Attrition = Yes)', fontsize=16)
    plt.xlabel(col, fontsize=12)
plt.tight_layout(pad=3.0)
plt.show()
```



In [339…

```python
# numerical_cols = data.select_dtypes(include=['int64', 'float64']).columns
```

In [290…

```
from scipy.stats import chi2_contingency

for col in categorical_cols:
    contingency_table = pd.crosstab(data[col], data['Attrition'])
    chi2, p, dof, expected = chi2_contingency(contingency_table)
    # p-value'u normal formata çevir
    formatted_p = f"{p:.20f}"
    print(f'{col} - p-value: {formatted_p}')
```

```
Attrition - p-value: 0.00000000000000000000
BusinessTravel - p-value: 0.00000560861447644993
Department - p-value: 0.00452560657447963286
EducationField - p-value: 0.00677398013902521211
Gender - p-value: 0.29057244902890855265
JobRole - p-value: 0.00000000000000275248
MaritalStatus - p-value: 0.00000000009455511060
OverTime - p-value: 0.00000000000000000001
```

In [291…

```
from scipy.stats import ttest_ind

for col in numerical_cols:
    group_yes = data[data['Attrition'] == 'Yes'][col]
    group_no = data[data['Attrition'] == 'No'][col]
    t_stat, p = ttest_ind(group_yes, group_no, nan_policy='omit')
    formatted_p = f"{p:.15f}"
    print(f'{col} - p-value: {formatted_p}')
```

```
Age - p-value: 0.000000000835631
DailyRate - p-value: 0.029858160660265
DistanceFromHome - p-value: 0.002793060080213
Education - p-value: 0.229315203322304
EnvironmentSatisfaction - p-value: 0.000071723385494
HourlyRate - p-value: 0.793134768994424
JobInvolvement - p-value: 0.000000567706536
JobLevel - p-value: 0.000000000067954
JobSatisfaction - p-value: 0.000070430667417
MonthlyIncome - p-value: 0.000000000714736
MonthlyRate - p-value: 0.561123598224301
NumCompaniesWorked - p-value: 0.095525262056519
PercentSalaryHike - p-value: 0.605612823889376
PerformanceRating - p-value: 0.911884042106900
RelationshipSatisfaction - p-value: 0.078713630484658
StockOptionLevel - p-value: 0.000000130101497
TotalWorkingYears - p-value: 0.000000000040619
TrainingTimesLastYear - p-value: 0.022578499737201
WorkLifeBalance - p-value: 0.014211054989021
YearsAtCompany - p-value: 0.000000231887161
YearsInCurrentRole - p-value: 0.000000000600319
YearsSinceLastPromotion - p-value: 0.205789959162494
YearsWithCurrManager - p-value: 0.000000001736987
```

Interpreting Variables with P < 0.05

According to the hypothesis test results, variables with $p < 0.05$ are factors that significantly affect the turnover status. These variables can be summarized as follows:

Categorical Variables

BusinessTravel:

Frequency of business travel is an important factor affecting turnover. For example, employees who travel frequently may have a higher tendency to leave.

Department:

The department in which the employee works has a significant effect on turnover.

EducationField:

A significant relationship was found between the employee's field of education and turnover.

JobRole:

Job role is an important factor on turnover.

MaritalStatus:

Single employees (or different marital statuses) may have higher turnover rates.

OverTime:

Turnover rates may be higher among employees who work overtime.

Numerical Variables

Age:

It may mean that younger employees are more likely to quit their jobs.

DistanceFromHome:

The distance between home and work may affect employees’ decision to quit their jobs.

EnvironmentSatisfaction:

Employees who are less satisfied with their work environment are more likely to quit their jobs.

JobInvolvement:

Employees with lower job commitment have higher quit rates.

JobSatisfaction:

Employees with lower job satisfaction have higher quit rates.

MonthlyIncome:

Salary level is an influential factor in quitting their jobs.

WorkLifeBalance:

Employees with lower work-life balance may be more likely to quit their jobs.

Conclusion

Take into account employee feedback to increase workplace satisfaction. Review overtime practices to improve work-life balance. Implement role changes or job development programs that will provide high job satisfaction. Review salary policies to make them competitive.