

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

df = pd.read_csv("HR_Analytics.csv")

print(df.head())
```

```

EmpID  Age  AgeGroup  Attrition  BusinessTravel  DailyRate  \
0  RM297   18   18-25      Yes      Travel_Rarely      230
1  RM302   18   18-25      No      Travel_Rarely      812
2  RM458   18   18-25      Yes  Travel_Frequently     1306
3  RM728   18   18-25      No      Non-Travel       287
4  RM829   18   18-25      Yes      Non-Travel       247

      Department  DistanceFromHome  Education  EducationField  ...  \
0  Research & Development           3          3  Life Sciences  ...
1             Sales              10          3    Medical     ...
2             Sales              5          3    Marketing     ...
3  Research & Development           5          2  Life Sciences  ...
4  Research & Development           8          1    Medical     ...

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

TotalWorkingYears  TrainingTimesLastYear  WorkLifeBalance  YearsAtCompany  \
0              0              2              3              0
1              0              2              3              0
2              0              3              3              0
3              0              2              3              0
4              0              0              3              0

YearsInCurrentRole  YearsSinceLastPromotion  YearsWithCurrManager
0              0              0              0.0
1              0              0              0.0
2              0              0              0.0
3              0              0              0.0
4              0              0              0.0

[5 rows x 38 columns]
```

#### #Data Preprocessing

```
df['Attrition'] = df['Attrition'].map({'Yes': 1, 'No': 0})

drop_cols = ['EmployeeNumber', 'EmployeeCount', 'Over18', 'StandardHours']
df.drop(columns=[col for col in drop_cols if col in df.columns], inplace=True)

X = pd.get_dummies(df.drop('Attrition', axis=1), drop_first=True)
y = df['Attrition']

print("Shape of features:", X.shape)
```

```
Shape of features: (1480, 1521)
```

```
print("Missing values per column:\n", X.isnull().sum())
```

```
X = X.dropna()
y = y[X.index]
```

```

Missing values per column:
Age              0
DailyRate        0
DistanceFromHome 0
Education        0
EnvironmentSatisfaction  ..
MaritalStatus_Single 0
```

```
SalarySlab_15k+      0
SalarySlab_5k-10k    0
SalarySlab_Upto 5k    0
OverTime_Yes         0
Length: 1521, dtype: int64
```

```
X_train, y_train, X_test, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

```
y = df['Attrition']
```

```
X = df.drop('Attrition', axis=1)
```

```
X = pd.get_dummies(X, drop_first=True)
```

```
X = X.dropna()
```

```
y = y.loc[X.index]
```

```
from sklearn.model_selection import train_test_split
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

```
from sklearn.linear_model import LogisticRegression
```

```
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
```

```
model = LogisticRegression(max_iter=1000)
```

```
model.fit(X_train, y_train)
```

```
y_pred = model.predict(X_test)
```

```
print("✅ Accuracy:", accuracy_score(y_test, y_pred))
```

```
print("✅ Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
```

```
print("✅ Classification Report:\n", classification_report(y_test, y_pred))
```



```
✅ Accuracy: 0.8641686182669789
```

```
✅ Confusion Matrix:
```

```
[[350  7]
 [ 51 19]]
```

```
✅ Classification Report:
```

	precision	recall	f1-score	support
No	0.87	0.98	0.92	357
Yes	0.73	0.27	0.40	70
accuracy			0.86	427
macro avg	0.80	0.63	0.66	427
weighted avg	0.85	0.86	0.84	427

```
/usr/local/lib/python3.11/dist-packages/sklearn/linear_model/_logistic.py:465: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

```
Increase the number of iterations (max_iter) or scale the data as shown in:
```

```
https://scikit-learn.org/stable/modules/preprocessing.html
```

```
Please also refer to the documentation for alternative solver options:
```

```
https://scikit-learn.org/stable/modules/linear\_model.html#logistic-regression
```

```
n_iter_i = _check_optimize_result(
```

```
# EDA Visualization Examples
```

```
# Attrition by Age
```

```
sns.histplot(data=df, x='Age', hue='Attrition', multiple='stack')
```

```
plt.title('Attrition by Age')
```

```
plt.show()
```

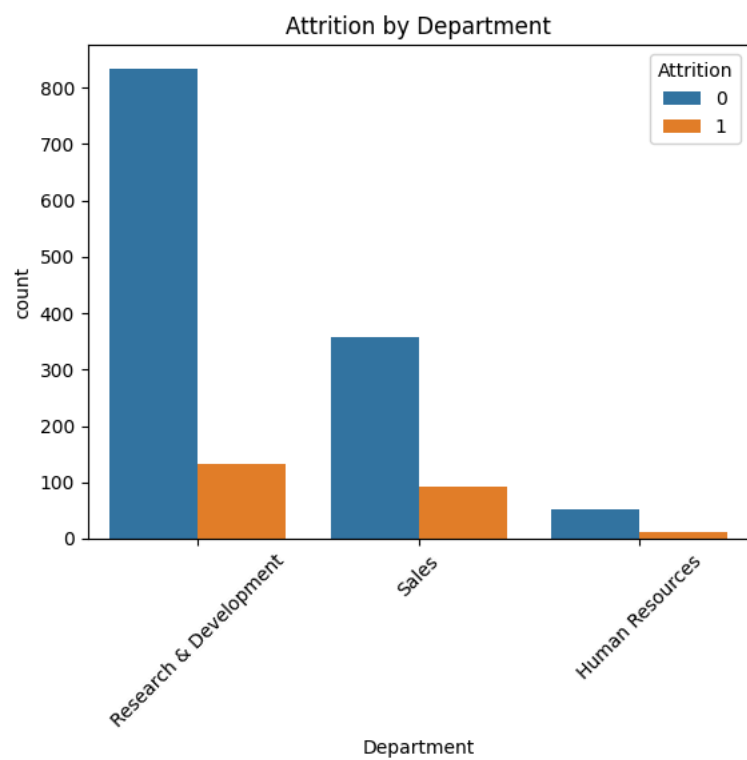
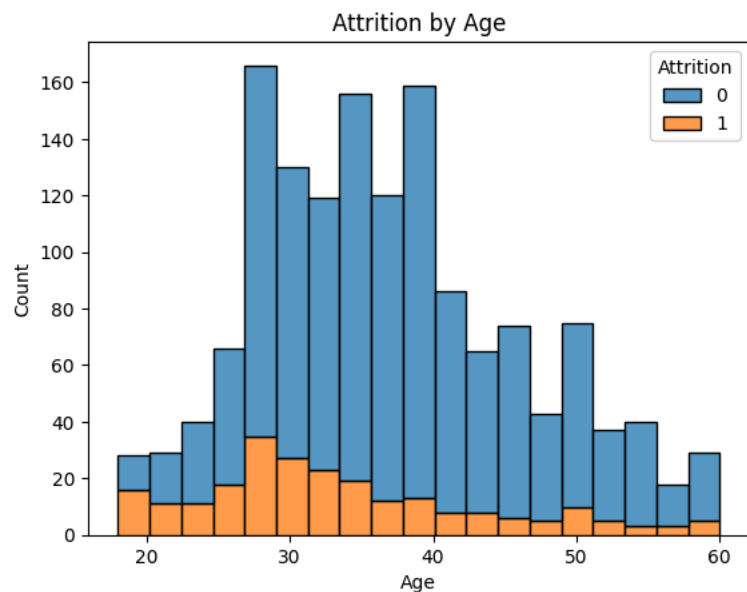
```
# Attrition by Department
```

```
sns.countplot(data=df, x='Department', hue='Attrition')
```

```
plt.title('Attrition by Department')
```

```
plt.xticks(rotation=45)
```

```
plt.show()
```



Start coding or [generate](#) with AI.