Sakshi Sawant (MSCIT-II/6957)

Practical 05

Aim - HR Analytics based on employee attrition factors.

```
[1] import pandas as pd

    # Sample employee data
    data = {
        'EmployeeID': [1, 2, 3, 4, 5],
        'HireDate': ['2022-01-01', '2022-05-15', '2022-07-20', '2022-09-01', '2022-11-10'],
        'TerminationDate': ['2023-01-10', '2023-04-22', None, '2023-08-05', None],
        'Department': ['Sales', 'IT', 'Sales', 'HR', 'IT'],
        'ReasonForLeaving': ['Resignation', 'Retirement', None, 'Layoff', None]
    }

    df = pd.DataFrame(data)
```

```
[2] # Convert dates to datetime format
    df['HireDate'] = pd.to_datetime(df['HireDate'])
    df['TerminationDate'] = pd.to_datetime(df['TerminationDate'])

    # Calculate the number of employees who left during the period
    employees_left = df['TerminationDate'].notna().sum()

    # Assuming the period is 1 year for simplicity, adjust as necessary
    period_start = pd.to_datetime('2023-01-01')
    period_end = pd.to_datetime('2023-12-31')
```

```
[3] # Calculate the average number of employees during the period
    employees_start = df[(df['HireDate'] <= period_start)].shape[0]
    employees_end = df[(df['HireDate'] <= period_end) & (df['TerminationDate'].isna() | (df['TerminationDate'] > period_end))].shape[0]
    average_employees = (employees_start + employees_end) / 2

    # Calculate turnover rate
    turnover_rate = (employees_left / average_employees) * 100
    print(f"Employee Turnover Rate: {turnover_rate:.2f}%")

    # Analyzing contributing factors
    turnover_by_department = df[df['TerminationDate'].notna()].groupby('Department').size()
    print("\nTurnover by Department:\n", turnover_by_department)

    turnover_reasons = df['ReasonForLeaving'].value_counts()
    print("\nReasons for Leaving:\n", turnover_reasons)
```

```
Employee Turnover Rate: 85.71%

Turnover by Department:
 Department
HR       1
IT       1
Sales    1
dtype: int64

Reasons for Leaving:
 ReasonForLeaving
Resignation    1
Retirement     1
Layoff         1
Name: count, dtype: int64
```

Python Code for Cox PH Model

```
[5] !pip install lifelines
```

```
Collecting lifelines
  Downloading lifelines-0.29.0-py3-none-a
Requirement already satisfied: numpy<2.0,
Requirement already satisfied: scipy>=1.7
Requirement already satisfied: pandas>=2.
```

```python
import pandas as pd
import numpy as np
from lifelines import KaplanMeierFitter, CoxPHFitter
import matplotlib.pyplot as plt

# Sample data preparation (assuming you have loaded your data into df)
df = pd.read_csv('/content/employee_data - employee_data.csv')

# Convert dates to datetime
df['HireDate'] = pd.to_datetime(df['HireDate'])
df['TerminationDate'] = pd.to_datetime(df['TerminationDate'])

# Create tenure feature (days worked)
df['Tenure'] = (df['TerminationDate'].fillna(pd.Timestamp('today')) - df['HireDate']).dt.days

# Create a 'Turnover' binary column (1 if left, 0 if still employed)
df['Turnover'] = df['TerminationDate'].notna().astype(int)

# One-hot encode categorical variables (e.g., Department, JobRole)
df_encoded = pd.get_dummies(df, columns=['Department', 'JobRole', 'ReasonForLeaving'], drop_first=True)

# Cox Proportional Hazards Model
cph = CoxPHFitter()

# Fit the model on the encoded data
cph.fit(df_encoded[['Tenure', 'Turnover'] + list(df_encoded.columns[6:])], duration_col='Tenure', event_col='Turnover')

# Print the summary of the model
cph.print_summary()
```

model	lifelines.CoxPHFitter
duration col	'Tenure'
event col	'Turnover'
baseline estimation	breslow
number of observations	200
number of events observed	57
partial log-likelihood	-213.32
time fit was run	2024-09-17 09:04:58 UTC

	coef	exp(coef)	se(coef)	coef lower 95%	coef upper 95%	exp(coef) lower 95%	exp(coef) upper 95%	cmp to	z	p	-log2(p)
Department_IT	0.21	1.23	0.49	-0.75	1.16	0.47	3.21	0.00	0.42	0.67	0.57
Department_Marketing	-0.52	0.59	0.41	-1.34	0.29	0.26	1.34	0.00	-1.26	0.21	2.27
Department_Sales	0.16	1.17	0.45	-0.73	1.04	0.48	2.84	0.00	0.35	0.73	0.46
JobRole_Developer	0.07	1.07	0.44	-0.80	0.93	0.45	2.54	0.00	0.15	0.88	0.18
JobRole_Executive	-0.05	0.95	0.47	-0.96	0.87	0.38	2.38	0.00	-0.10	0.92	0.12
JobRole_Manager	-0.02	0.98	0.56	-1.11	1.08	0.33	2.94	0.00	-0.03	0.98	0.04
JobRole_Salesperson	0.58	1.79	0.46	-0.31	1.48	0.73	4.40	0.00	1.27	0.20	2.30
ReasonForLeaving_Layoff	3.90	49.46	0.51	2.91	4.89	18.33	133.43	0.00	7.70	<0.005	46.11
ReasonForLeaving_Resignation	4.62	101.40	0.53	3.57	5.67	35.54	289.33	0.00	8.63	<0.005	57.23
ReasonForLeaving_Retirement	3.76	42.95	0.50	2.77	4.75	16.01	115.21	0.00	7.47	<0.005	43.50
ReasonForLeaving_Termination	4.07	58.43	0.47	3.14	4.99	23.12	147.64	0.00	8.60	<0.005	56.81

Concordance	0.87
Partial AIC	448.65
log-likelihood ratio test	155.49 on 11 df
-log2(p) of ll-ratio test	89.52

```python
# Plot the coefficients
cph.plot()
plt.title('Cox Proportional Hazards Model Coefficients')
plt.show()
```