In [ ]:
```
!pip install -qU torch matplotlib seaborn pandas ydata-profiling scikit-learn ipywidgets
```

In [ ]:
```
import pandas as pd

df = pd.read_csv("./archive/WA_Fn-UseC_-HR-Employee-Attrition.csv")
```

In [ ]:
```
from IPython.display import display, Markdown
pd.set_option("display.max_colwidth", None)
```

In [ ]:
```
import ydata_profiling

ydata_profiling.ProfileReport(df)
```

```
Summarize dataset:   0%|          | 0/5 [00:00<?, ?it/s]
Generate report structure:   0%|          | 0/1 [00:00<?, ?it/s]
Render HTML:   0%|          | 0/1 [00:00<?, ?it/s]
```

m	1024	4.2%
Other values (10)	7425	30.5%

Most occurring blocks

Value	Count	Frequency (%)
(unknown)	24317	100.0%

Most frequent character per block

(unknown)

Value	Count	Frequency (%)
e	5377	22.1%
	1985	8.2%
s	1533	6.3%
a	1470	6.0%
l	1407	5.8%
R	1024	4.2%
r	1024	4.2%
c	1024	4.2%

n	1024	4.2%
m	1024	4.2%

Out[ ]:

- we would like to predict the attrition rate so we will target 'Attrition'
- we will also one-hot-encode some of these features

In [ ]:
```
df.Gender.value_counts()
```

Out[ ]:
```
Gender
Male      882
Female    588
Name: count, dtype: int64
```

In [ ]:
```
# binary encoding attrition, Over18, and OverTime

df['Attrition'] = df['Attrition'].apply(lambda x: 1 if x == 'Yes' else 0)
df['Over18'] = df['Over18'].apply(lambda x: 1 if x == 'Y' else 0)
df['OverTime'] = df['OverTime'].apply(lambda x: 1 if x == 'Yes' else 0)
df['Gender'] = df['Gender'].apply(lambda x: 1 if x == 'Male' else 0)
```

In [ ]:
```
df
```

Out[ ]:

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	Empl
0	41	1	Travel_Rarely	1102	Sales	1	2	Life Sciences	
1	49	0	Travel_Frequently	279	Research & Development	8	1	Life Sciences	
2	37	1	Travel_Rarely	1373	Research & Development	2	2	Other	
3	33	0	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	
4	27	0	Travel_Rarely	591	Research & Development	2	1	Medical	
...	...	...	...	...	...	...	...	...	
1465	36	0	Travel_Frequently	884	Research & Development	23	2	Medical	
1466	39	0	Travel_Rarely	613	Research & Development	6	1	Medical	
1467	27	0	Travel_Rarely	155	Research & Development	4	3	Life Sciences	
1468	49	0	Travel_Frequently	1023	Sales	2	3	Medical	
1469	34	0	Travel_Rarely	628	Research & Development	8	3	Medical	

1470 rows × 35 columns

In [ ]:
```
# One hot encoding: Department, EducationField, JobRole, MaritalStatus, Business travel, EducationField

df = df.join(pd.get_dummies(df['BusinessTravel'])).drop('BusinessTravel', axis=1)
df = df.join(pd.get_dummies(df['Department'], prefix='Department')).drop('Department', axis=1)
```

```
df = df.join(pd.get_dummies(df['EducationField'], prefix='EducationField')).drop('EducationField', axis=1)
df = df.join(pd.get_dummies(df['JobRole'], prefix='JobRole')).drop('JobRole', axis=1)
df = df.join(pd.get_dummies(df['MaritalStatus'], prefix='MaritalStatus')).drop('MaritalStatus', axis=1)
```

In [ ]:
```
df
```

Out[ ]:

	Age	Attrition	DailyRate	DistanceFromHome	Education	EmployeeCount	EmployeeNumber	EnvironmentSatisf
0	41	1	1102	1	2	1	1	
1	49	0	279	8	1	1	2	
2	37	1	1373	2	2	1	4	
3	33	0	1392	3	4	1	5	
4	27	0	591	2	1	1	7	
...	...	...	...	...	...	...	...	
1465	36	0	884	23	2	1	2061	
1466	39	0	613	6	1	1	2062	
1467	27	0	155	4	3	1	2064	
1468	49	0	1023	2	3	1	2065	
1469	34	0	628	8	3	1	2068	

1470 rows × 54 columns

In [ ]:
```
df = df.map(lambda x: 1 if x is True else 0 if x is False else x)
df
```

Out[ ]:

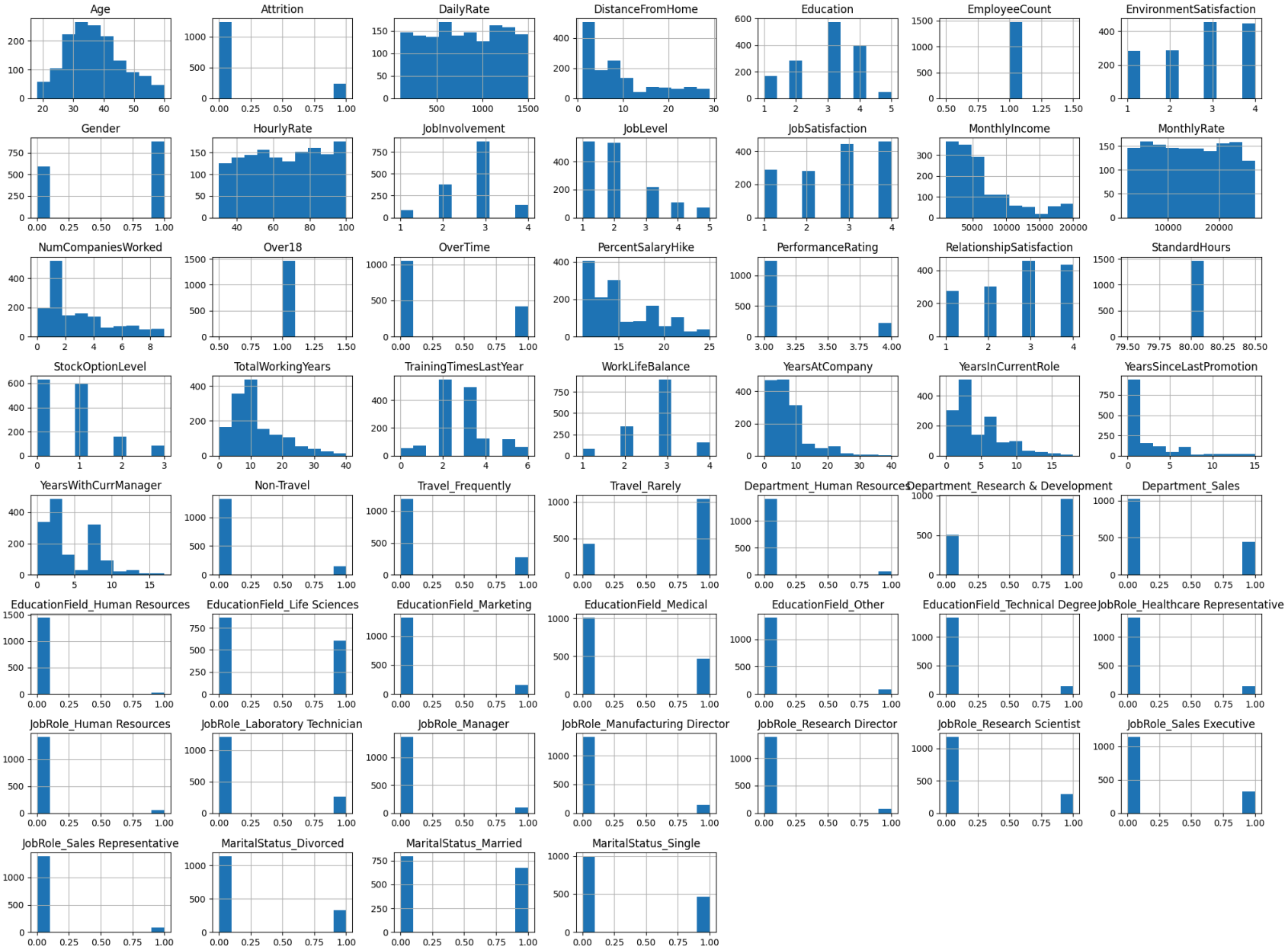
	Age	Attrition	DailyRate	DistanceFromHome	Education	EmployeeCount	EmployeeNumber	EnvironmentSatisf
0	41	1	1102	1	2	1	1	
1	49	0	279	8	1	1	2	
2	37	1	1373	2	2	1	4	
3	33	0	1392	3	4	1	5	
4	27	0	591	2	1	1	7	
...	...	...	...	...	...	...	...	
1465	36	0	884	23	2	1	2061	
1466	39	0	613	6	1	1	2062	
1467	27	0	155	4	3	1	2064	
1468	49	0	1023	2	3	1	2065	
1469	34	0	628	8	3	1	2068	

1470 rows × 54 columns

In [ ]:
```
df = df.drop('EmployeeNumber', axis=1)
```

In [ ]:
```
import matplotlib.pyplot as plt
%matplotlib inline

df.hist(figsize=(20, 15))
plt.tight_layout()
plt.show()
```



In [ ]:
```
df = df.drop(['EmployeeCount', 'Over18', 'StandardHours'], axis=1)
```

In [ ]:
```
df
```

Out[ ]:

	Age	Attrition	DailyRate	DistanceFromHome	Education	EnvironmentSatisfaction	Gender	HourlyRate	JobInv
0	41	1	1102	1	2	2	0	94	
1	49	0	279	8	1	3	1	61	
2	37	1	1373	2	2	4	1	92	
3	33	0	1392	3	4	4	0	56	
4	27	0	591	2	1	1	1	40	
...	...	...	...	...	...	...	...	...	
1465	36	0	884	23	2	3	1	41	
1466	39	0	613	6	1	4	1	42	
1467	27	0	155	4	3	2	1	87	
1468	49	0	1023	2	3	4	1	63	
1469	34	0	628	8	3	2	1	82	

1470 rows × 50 columns

In [ ]:
```
# model training

from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split

X, y = df.drop('Attrition', axis=1), df['Attrition']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

model = RandomForestClassifier(n_jobs=-1)

model.fit(X_train, y_train)
```

Out[ ]:
```
RandomForestClassifier
RandomForestClassifier(n_jobs=-1)
```

In [ ]:
```
model.score(X_test, y_test)
```

Out[ ]:
```
0.8809523809523809
```

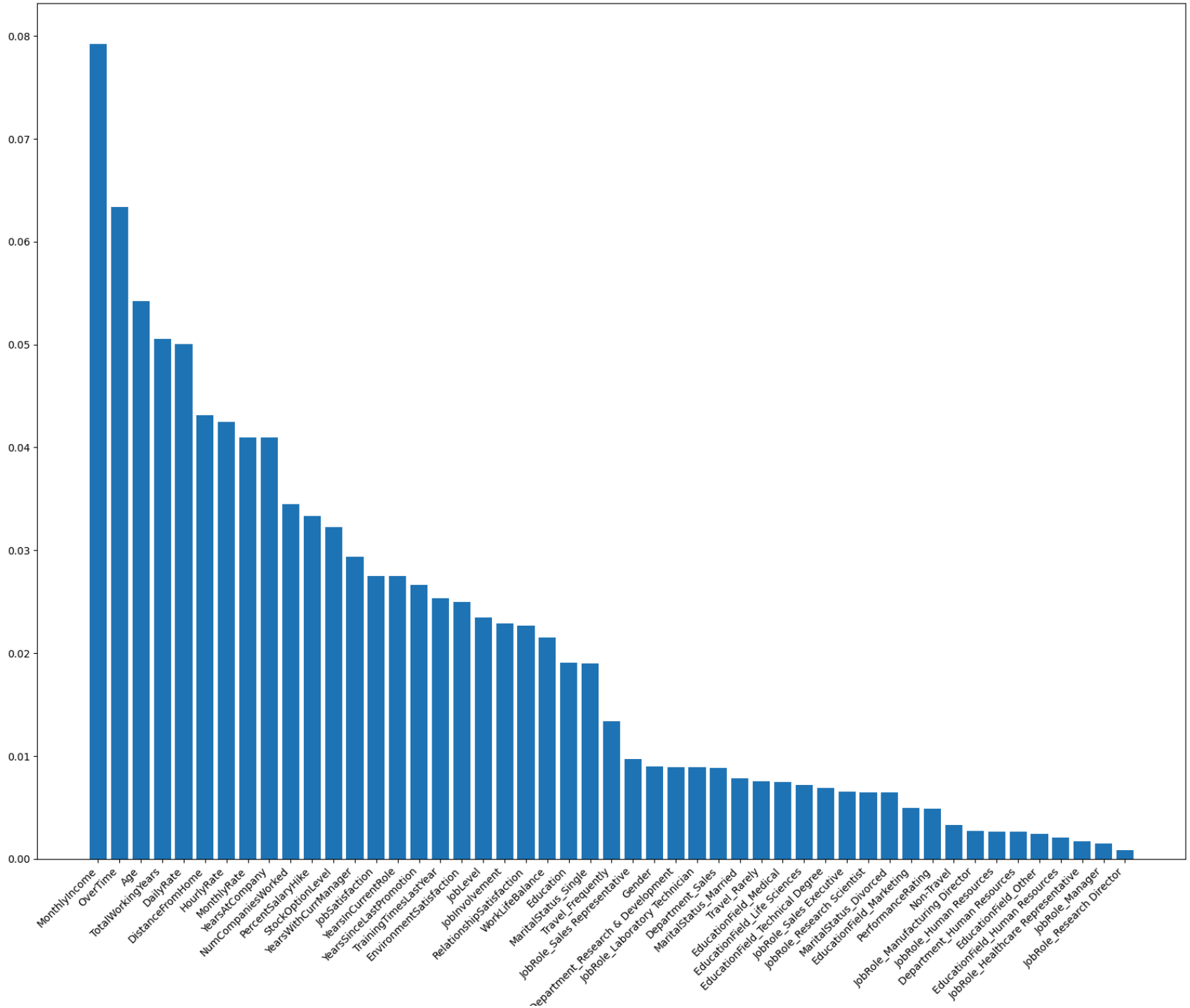
In [ ]:
```
# what factors are important in predicting attrition
attrition_factors = dict(sorted(zip(model.feature_names_in_, model.feature_importances_), key=lambda x: x[
```

In [ ]:
```
plt.figure(figsize=(20, 15))
plt.bar(attrition_factors.keys(), attrition_factors.values())
plt.xticks(rotation=45, ha='right')
```

```
Out[ ]: ([0,
          1,
          2,
          3,
          4,
          5,
          6,
          7,
          8,
          9,
          10,
          11,
          12,
          13,
          14,
          15,
          16,
          17,
          18,
          19,
          20,
          21,
          22,
          23,
          24,
          25,
          26,
          27,
          28,
          29,
          30,
          31,
          32,
          33,
          34,
          35,
          36,
          37,
          38,
          39,
          40,
          41,
```

```
 42,
 43,
 44,
 45,
 46,
 47,
 48],
[Text(0, 0, 'MonthlyIncome'),
 Text(1, 0, 'OverTime'),
 Text(2, 0, 'Age'),
 Text(3, 0, 'TotalWorkingYears'),
 Text(4, 0, 'DailyRate'),
 Text(5, 0, 'DistanceFromHome'),
 Text(6, 0, 'HourlyRate'),
 Text(7, 0, 'MonthlyRate'),
 Text(8, 0, 'YearsAtCompany'),
 Text(9, 0, 'NumCompaniesWorked'),
 Text(10, 0, 'PercentSalaryHike'),
 Text(11, 0, 'StockOptionLevel'),
 Text(12, 0, 'YearsWithCurrManager'),
 Text(13, 0, 'JobSatisfaction'),
 Text(14, 0, 'YearsInCurrentRole'),
 Text(15, 0, 'YearsSinceLastPromotion'),
 Text(16, 0, 'TrainingTimesLastYear'),
 Text(17, 0, 'EnvironmentSatisfaction'),
 Text(18, 0, 'JobLevel'),
 Text(19, 0, 'JobInvolvement'),
 Text(20, 0, 'RelationshipSatisfaction'),
 Text(21, 0, 'WorkLifeBalance'),
 Text(22, 0, 'Education'),
 Text(23, 0, 'MaritalStatus_Single'),
 Text(24, 0, 'Travel_Frequently'),
 Text(25, 0, 'JobRole_Sales Representative'),
 Text(26, 0, 'Gender'),
 Text(27, 0, 'Department_Research & Development'),
 Text(28, 0, 'JobRole_Laboratory Technician'),
 Text(29, 0, 'Department_Sales'),
 Text(30, 0, 'MaritalStatus_Married'),
 Text(31, 0, 'Travel_Rarely'),
 Text(32, 0, 'EducationField_Medical'),
 Text(33, 0, 'EducationField_Life Sciences'),
 Text(34, 0, 'EducationField_Technical Degree'),
```

```
                Text(35, 0, 'JobRole_Sales Executive'),
                Text(36, 0, 'JobRole_Research Scientist'),
                Text(37, 0, 'MaritalStatus_Divorced'),
                Text(38, 0, 'EducationField_Marketing'),
                Text(39, 0, 'PerformanceRating'),
                Text(40, 0, 'Non-Travel'),
                Text(41, 0, 'JobRole_Manufacturing Director'),
                Text(42, 0, 'JobRole_Human Resources'),
                Text(43, 0, 'Department_Human Resources'),
                Text(44, 0, 'EducationField_Other'),
                Text(45, 0, 'EducationField_Human Resources'),
                Text(46, 0, 'JobRole_Healthcare Representative'),
                Text(47, 0, 'JobRole_Manager'),
                Text(48, 0, 'JobRole_Research Director')])
```


0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
MonthlyIncome
OverTime
Age
TotalWorkingYears
DailyRate
DistanceFromHome
HourlyRate
MonthlyRate
YearsAtCompany
NumCompaniesWorked
PercentSalaryHike
StockOptionLevel
YearsWithCurrManager
JobSatisfaction
YearsInCurrentRole
YearsSinceLastPromotion
TrainingTimesLastYear
EnvironmentSatisfaction
JobLevel
JobInvolvement
RelationshipSatisfaction
WorkLifeBalance
Education
MaritalStatus_Single
Travel_Frequently
JobRole_Sales Representative
Department_Research & Development
Gender
JobRole_Laboratory Technician
Department_Sales
MaritalStatus_Married
Travel_Rarely
EducationField_Medical
EducationField_Life Sciences
EducationField_Technical Degree
JobRole_Sales Executive
JobRole_Research Scientist
MaritalStatus_Divorced
EducationField_Marketing
PerformanceRating
Non-Travel
JobRole_Manufacturing Director
JobRole_Human Resources
Department_Human Resources
EducationField_Other
EducationField_Human Resources
JobRole_Healthcare Representative
JobRole_Manager
JobRole_Research Director

In [ ]: