

Random Forest and Decision Tree of Attrition Data set.

```
dataset4=pd.read_csv("Attrition Rate Analysis.csv")
```

```
dataset4.columns
```

```
Out[26]:
```

```
Index(['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome',  
      'Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gender',  
      'JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome',  
      'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours',  
      'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear',  
      'YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager'],  
      dtype='object')
```

```
from sklearn import preprocessing
```

```
le=preprocessing.LabelEncoder()
```

```
dataset4["Attrition"]=le.fit_transform(dataset4["Attrition"])
```

```
dataset4["BusinessTravel"]=le.fit_transform(dataset4["BusinessTravel"])
```

```
dataset4["Department"]=le.fit_transform(dataset4["Department"])
```

```
dataset4["EducationField"]=le.fit_transform(dataset4["EducationField"])
```

```
dataset4["Gender"]=le.fit_transform(dataset4["Gender"])
```

```
dataset4["MaritalStatus"]=le.fit_transform(dataset4["MaritalStatus"])
```

```
dataset4["JobRole"]=le.fit_transform(dataset4["JobRole"])
```

```
dataset4.columns
```

```
Out[36]:
```

```
Index(['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome',  
      'Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gender',  
      'JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome',  
      'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours',  
      'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear',  
      'YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager'],  
      dtype='object')
```

```
dataset5=dataset4.drop(['EmployeeCount', 'EmployeeID','Over18','StandardHours'],axis=1)
```

```
dataset5.columns
```

```
Out[38]:
```

```
Index(['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome',  
      'Education', 'EducationField', 'Gender', 'JobLevel', 'JobRole',  
      'MaritalStatus', 'MonthlyIncome', 'NumCompaniesWorked',  
      'PercentSalaryHike', 'StockOptionLevel', 'TotalWorkingYears',  
      'TrainingTimesLastYear', 'YearsAtCompany', 'YearsSinceLastPromotion',  
      'YearsWithCurrManager'],  
      dtype='object')
```

```
from sklearn.ensemble import RandomForestClassifier
```

```
dataset6=dataset5.dropna()
```

```
dataset7=dataset6.drop_duplicates()
```

```
rf_model=RandomForestClassifier(n_estimators=1000,max_features=2,oob_score=True)
```

```
features=['Age', 'BusinessTravel', 'Department', 'DistanceFromHome','Education',  
'EducationField', 'Gender', 'JobLevel', 'JobRole','MaritalStatus', 'MonthlyIncome',  
'NumCompaniesWorked','PercentSalaryHike', 'StockOptionLevel',  
'TotalWorkingYears','TrainingTimesLastYear', 'YearsAtCompany',  
'YearsSinceLastPromotion','YearsWithCurrManager']
```

```
rf_model.fit(X=dataset7[features],y=dataset7["Attrition"])
```

```
Out[44]:
```

```
RandomForestClassifier(bootstrap=True, ccp_alpha=0.0, class_weight=None,  
                        criterion='gini', max_depth=None, max_features=2,  
                        max_leaf_nodes=None, max_samples=None,  
                        min_impurity_decrease=0.0, min_impurity_split=None,  
                        min_samples_leaf=1, min_samples_split=2,  
                        min_weight_fraction_leaf=0.0, n_estimators=1000,  
                        n_jobs=None, oob_score=True, random_state=None,  
                        verbose=0, warm_start=False)
```

```
print(rf_model.oob_score_)
```

```
0.8401360544217688
```

```
for features,imp in zip(features,rf_model.feature_importances_):
```

```
    print(features,imp);
```

```
Age 0.09752674191339729
```

```
BusinessTravel 0.027563094213474554
```

```
Department 0.02644145328044662
```

```
DistanceFromHome 0.0673774092636159
```

Education 0.04107623410409063
EducationField 0.04143587997346895
Gender 0.018613315741411385
JobLevel 0.03691552620952532
JobRole 0.05430078292962142
MaritalStatus 0.04039942500865896
MonthlyIncome 0.09321661099948114
NumCompaniesWorked 0.05645815033877504
PercentSalaryHike 0.06568672450120092
StockOptionLevel 0.034400277704681875
TotalWorkingYears 0.08691848144611461
TrainingTimesLastYear 0.04451288378473802
YearsAtCompany 0.06906332241475244
YearsSinceLastPromotion 0.04432488543953697
YearsWithCurrManager 0.053768800733007896

from sklearn import tree

tree_model=tree.DecisionTreeClassifier(max_depth=6,max_leaf_nodes=10)

predictors=pd.DataFrame([dataset7["TotalWorkingYears"],dataset7["Age"],dataset7["MonthlyIncome"]])

predictors=pd.DataFrame([dataset7["TotalWorkingYears"],dataset7["Age"],dataset7["MonthlyIncome"]]).T

tree_model.fit(X=predictors,y=dataset7["Attrition"])

Out[54]:

DecisionTreeClassifier(ccp_alpha=0.0, class_weight=None, criterion='gini',
max_depth=6, max_features=None, max_leaf_nodes=10,
min_impurity_decrease=0.0, min_impurity_split=None,

```

min_samples_leaf=1, min_samples_split=2,
min_weight_fraction_leaf=0.0, presort='deprecated',
random_state=None, splitter='best')

```

with open("Dtree2.dot","w")as f:

```

f=tree.export_graphviz(tree_model,feature_names=["TotalWorkingYears","Age","MonthlyIn
come"],out_file=f);

```

