COURSE PROJECT

DATA MINING

EMPLOYEE ATTRITION DATA ANALYTICS

INTRODUCTION

Our project is based on a comprehensive HR Analytics dataset referred by Kaggle Datasets (https://www.kaggle.com/datasets). The dataset comprises information on 1,470 employees, captured across 35 distinct attributes. It holds a comprehensive range of data, including demographic details like age and gender, alongside specific job-related characteristics such as department, role, and salary levels. Further, it delves into aspects of work conditions, recording metrics on business travel, commuting distance, and daily work rates. A significant portion of the dataset is dedicated to evaluating employee satisfaction and engagement through various measures such as job satisfaction, work-life balance, and environment satisfaction. Additionally, it provides details into career progression by documenting years of service, role tenure, and promotion history. Benefits and compensation are also detailed, highlighting elements like stock options and salary increments. The dataset's thoroughness in covering these dimensions makes it a robust resource for analyzing workforce dynamics, particularly in areas related to employee retention and career development, with each attribute being fully populated, ensuring a comprehensive base for analytical assessments.

MAIN CHAPTER

Step 1 - Develop understanding of purpose of DM project

Our project, HR Analytics, is focused on attrition prediction, the initial step involves defining the objectives clearly. This data set involves supervised learning where the response is categorical with classes. Here, the aim is to leverage data mining techniques to predict potential attrition cases within the organization. Understanding the purpose entails identifying the key stakeholders, such as HR personnel and people managers, and outlining their needs and expectations. By clearly defining the project's goals, the subsequent steps can be aligned effectively to address the challenge of employee attrition.

Step 2 - Obtain data for analysis

The next step involves gathering relevant data necessary for analysis. In this context, the data source Employee Attrition.csv includes employee demographics, performance metrics, salary information, and historical attrition records (like Department, Business Travel, Education Level) related to employees. Depending on the size and complexity of the dataset, random sampling techniques might be employed to ensure representativeness and manage computational resources effectively. This step is crucial as the quality and comprehensiveness of the data significantly influences the accuracy and reliability of the predictive models.

Step 3 - Explore, clean, and preprocess data

Once the data is collected, it undergoes thorough exploration, cleaning, and preprocessing. This involves identifying and handling missing values, removing duplicates, and addressing inconsistencies in the data. Additionally, preprocessing techniques such as normalization and feature scaling may be applied to prepare the data for further analysis. The data types of some of the predictors are objects, we need to convert them into dummy variables to further use for the analysis.

a. **Size of Data Frame**

The number of rows is 1470 and the number of columns is 35 in the data set Employee Attrition.csv.

```
Size of the DataFrame: (1470, 35)
```

b. The initial five rows of the dataset

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	1	1
1	49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences	1	2
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	1	4
3	33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	1	5
4	27	No	Travel_Rarely	591	Research & Development	2	1	Medical	1	7

c) **Original and Updated Column Names**

```
Original Columun Names : Index(['Age', 'Attrition', 'BusinessTravel', 'DailyRate', 'Department',
       'DistanceFromHome', 'Education', 'EducationField', 'EmployeeCount',
       'EmployeeNumber', 'EnvironmentSatisfaction', 'Gender', 'HourlyRate',
       'JobInvolvement', 'JobLevel', 'JobRole', 'JobSatisfaction',
       'MaritalStatus', 'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked',
       'Over18', 'OverTime', 'PercentSalaryHike', 'PerformanceRating',
       'RelationshipSatisfaction', 'StandardHours', 'StockOptionLevel',
       'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance',
       'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion',
       'YearsWithCurrManager'],
      dtype='object')
Single Columun Names : Index(['Age', 'Attrition', 'BusinessTravel', 'DailyRate', 'Department',
       'DistanceFromHome', 'Education', 'EducationField', 'EmployeeCount',
       'EmployeeNumber', 'EnvironmentSatisfaction', 'Gender', 'HourlyRate',
       'JobInvolvement', 'JobLevel', 'JobRole', 'JobSatisfaction',
       'MaritalStatus', 'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked',
       'Over18', 'OverTime', 'PercentSalaryHike', 'PerformanceRating',
       'RelationshipSatisfaction', 'StandardHours', 'StockOptionLevel',
       'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance',
       'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion',
       'YearsWithCurrManager'],
      dtype='object')
```

d) **Original Data Types**

```
The Data Type of Columns: Age                        int64
Attrition                 object
BusinessTravel            object
DailyRate                  int64
Department                object
DistanceFromHome           int64
Education                  int64
EducationField            object
EmployeeCount              int64
EmployeeNumber             int64
EnvironmentSatisfaction    int64
Gender                    object
HourlyRate                 int64
JobInvolvement             int64
JobLevel                   int64
JobRole                   object
JobSatisfaction            int64
MaritalStatus             object
MonthlyIncome              int64
MonthlyRate                int64
NumCompaniesWorked         int64
Over18                    object
OverTime                  object
PercentSalaryHike          int64
PerformanceRating          int64
```

e) **The data set contains 1470 rows and 35 columns (numerical and 9 categorical and) out of which some are irrelevant predictors.**

```
Index(['Age', 'Attrition', 'BusinessTravel', 'DailyRate', 'Department',
       'DistanceFromHome', 'Education', 'EducationField',
       'EnvironmentSatisfaction', 'Gender', 'HourlyRate', 'JobInvolvement',
       'JobLevel', 'JobRole', 'JobSatisfaction', 'MaritalStatus',
       'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked', 'OverTime',
       'PercentSalaryHike', 'PerformanceRating', 'RelationshipSatisfaction',
       'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear',
       'WorkLifeBalance', 'YearsAtCompany', 'YearsInCurrentRole',
       'YearsSinceLastPromotion', 'YearsWithCurrManager'],
      dtype='object')
```

f) **Data Types of updated predictors:**

```
Age                         int64
DailyRate                   int64
DistanceFromHome            int64
Education                   int64
EnvironmentSatisfaction     int64
HourlyRate                  int64
JobInvolvement              int64
JobLevel                    int64
JobSatisfaction             int64
MonthlyIncome               int64
MonthlyRate                 int64
NumCompaniesWorked          int64
PercentSalaryHike           int64
PerformanceRating           int64
RelationshipSatisfaction    int64
StockOptionLevel            int64
TotalWorkingYears           int64
TrainingTimesLastYear       int64
WorkLifeBalance             int64
YearsAtCompany              int64
YearsInCurrentRole          int64
YearsSinceLastPromotion     int64
YearsWithCurrManager        int64
```

g) **Updated columns using dummy variables**

```
Index(['Age', 'DailyRate', 'DistanceFromHome', 'Education',
       'EnvironmentSatisfaction', 'HourlyRate', 'JobInvolvement', 'JobLevel',
       'JobSatisfaction', 'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked',
       'PercentSalaryHike', 'PerformanceRating', 'RelationshipSatisfaction',
       'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear',
       'WorkLifeBalance', 'YearsAtCompany', 'YearsInCurrentRole',
       'YearsSinceLastPromotion', 'YearsWithCurrManager', 'Attrition_Yes',
       'BusinessTravel_Travel_Frequently', 'BusinessTravel_Travel_Rarely',
       'Department_Research & Development', 'Department_Sales',
       'EducationField_Life Sciences', 'EducationField_Marketing',
       'EducationField_Medical', 'EducationField_Other',
       'EducationField_Technical Degree', 'Gender_Male',
       'JobRole_Human Resources', 'JobRole_Laboratory Technician',
       'JobRole_Manager', 'JobRole_Manufacturing Director',
       'JobRole_Research Director', 'JobRole_Research Scientist',
       'JobRole_Sales Executive', 'JobRole_Sales Representative',
       'MaritalStatus_Married', 'MaritalStatus_Single', 'OverTime_Yes'],
      dtype='object')
```

h) **The count of missing values in each column**

```
The Total Number of Null Values in Each Column:
 Age                                0
DailyRate                           0
DistanceFromHome                    0
Education                           0
EnvironmentSatisfaction             0
HourlyRate                          0
JobInvolvement                      0
JobLevel                            0
JobSatisfaction                     0
MonthlyIncome                       0
MonthlyRate                         0
NumCompaniesWorked                  0
PercentSalaryHike                   0
PerformanceRating                   0
RelationshipSatisfaction            0
StockOptionLevel                    0
TotalWorkingYears                   0
TrainingTimesLastYear               0
WorkLifeBalance                     0
YearsAtCompany                      0
```

i) **Descriptive statistics of the Data Frame**

	Age	DailyRate	DistanceFromHome	Education	EnvironmentSatisfaction	HourlyRate	JobInvolvement	JobLevel	JobSatisfaction	MonthlyIncome
count	1470.00	1470.00	1470.00	1470.00	1470.00	1470.00	1470.00	1470.00	1470.00	1470.00
mean	36.92	802.49	9.19	2.91	2.72	65.89	2.73	2.06	2.73	6502.93
std	9.14	403.51	8.11	1.02	1.09	20.33	0.71	1.11	1.10	4707.96
min	18.00	102.00	1.00	1.00	1.00	30.00	1.00	1.00	1.00	1009.00
25%	30.00	465.00	2.00	2.00	2.00	48.00	2.00	1.00	2.00	2911.00
50%	36.00	802.00	7.00	3.00	3.00	66.00	3.00	2.00	3.00	4919.00
75%	43.00	1157.00	14.00	4.00	4.00	83.75	3.00	3.00	4.00	8379.00
max	60.00	1499.00	29.00	5.00	4.00	100.00	4.00	5.00	4.00	19999.00

Step 5 - Determine Data Mining tasks

Given the problem of attrition prediction, the primary data mining task is classification, where the objective is to classify employees into two categories: Whether the employee is staying or leaving the company.

By framing the problem as a classification task, appropriate machine learning algorithms and evaluation metrics can be selected to build predictive models effectively.

Step 6 - Partition data (for supervised task)

To train and evaluate the predictive models, the dataset is partitioned into training and validation sets with the training partition to be 60% and Validation partition as 40%. The training set is used to train the models, and the validation set is used to tune hyperparameters and assess model performance during training. This partitioning ensures unbiased model evaluation and helps prevent overfitting.

```
Training set - Features: (882, 44)
Validation set - Features: (588, 44)
```

Step 7 – Techniques

Logistic Model

Using LogisticRegression() Function for Multiple Predictors Logistic Regression

a) The intercept is 0.302 and the coefficients are as mentioned din the picture below:

```
Parameters of Logistic Regresion Model with Multiple Predictors
Intercept: 0.302
Coefficients for Predictors
           Age  DailyRate  DistanceFromHome  Education  \
Coeff: -0.019       -0.0             0.041      0.119

         EnvironmentSatisfaction  HourlyRate  JobInvolvement  JobLevel  \
Coeff:                    -0.446       0.008          -0.399     0.023

         JobSatisfaction  MonthlyIncome  ...  JobRole_Laboratory Technician  \
Coeff:            -0.253           -0.0  ...                           0.86

         JobRole_Manager  JobRole_Manufacturing Director  \
Coeff:             0.035                          -0.311

         JobRole_Research Director  JobRole_Research Scientist  \
Coeff:                     -0.072                       -0.27

         JobRole_Sales Executive  JobRole_Sales Representative  \
Coeff:                    -0.252                         0.599

         MaritalStatus_Married  MaritalStatus_Single  OverTime_Yes
Coeff:                   0.066                 0.746         1.832

[1 rows x 44 columns]
```

The model equation is

Logit = 0.302 - 0.019Age -0.0DailyRate +.....+0.746MaritalStatus_Single +

1.832OverTime_Yes

b) **Predictions and Probabilities Analysis using Multiple Predictors Logistic Regression Model on Validation Set**

Classification for Validation Partition

	Actual	Classification	p(0)	p(1)
1291	1	0	0.9732	0.0268
1153	1	1	0.2268	0.7732
720	1	1	0.4027	0.5973
763	0	0	0.5958	0.4042
976	0	0	0.8812	0.1188
724	0	0	0.9918	0.0082
314	0	0	0.8303	0.1697
258	0	0	0.9390	0.0610
442	0	0	0.8941	0.1059
1393	0	0	0.9807	0.0193
894	0	0	0.9996	0.0004
435	1	0	0.8300	0.1700
952	1	1	0.3724	0.6276
236	1	0	0.8596	0.1404
1170	0	0	0.9189	0.0811
1295	0	0	0.7611	0.2389
826	0	0	0.9675	0.0325
453	1	1	0.3511	0.6489
1230	0	0	0.8143	0.1857
702	0	0	0.9222	0.0778

Most of the first 20 validation records are correctly classified as 1 (Employee Stays). A few of the first 20 records are misclassified. They have the employee attrition as 0 (Employee Leaves), but the logistic regression model misclassified them as 0, because the probability of 0.

c) Confusion matrix for logistic model: This shows the accuracy of the number of the correct actuals of 0s and 1s. Here the accuracy seems to be 86.05% for the validation partition which has the misclassification of 13.95%.

```
Training Partition
Confusion Matrix (Accuracy 0.9036)

       Prediction
Actual   0   1
     0 740   7
     1  78  57

Validation Partition
Confusion Matrix (Accuracy 0.8605)

       Prediction
Actual   0   1
     0 469  17
     1  65  37
```

Ordinal Logistic Model:

a) **Development and Analysis of Ordinal Logistic Regression Model**

```
Ordinal Logistic Regression
  Intercepts [-0.019]
  Coefficients [-0.002 -0.     0.021  0.089 -0.335  0.012 -0.209 -0.002 -0.177 -0.
  0.     0.156  0.013  0.035 -0.041 -0.277 -0.077 -0.159 -0.112  0.112
 -0.181  0.239 -0.221  0.137 -0.009 -0.146  0.152 -0.008  0.12  -0.143
 -0.038  0.055  0.044  0.018  0.071  0.002 -0.059 -0.015 -0.078  0.045
  0.098 -0.106  0.224  0.395]


Classification for First 10 Records in Validation Data Set
      Actual  Classification    P(0)    P(1)
1291       1               0  0.9699  0.0301
1153       1               0  0.5435  0.4565
720        1               0  0.5996  0.4004
763        0               0  0.6322  0.3678
976        0               0  0.9270  0.0730
724        0               0  0.9482  0.0518
314        0               0  0.8915  0.1085
258        0               0  0.6735  0.3265
442        0               0  0.7226  0.2774
1393       0               0  0.9902  0.0098
```

b) Confusion matrix for Ordinal Logistic model:

```
Training Partition for Ordinal Logistic Model
Confusion Matrix (Accuracy 0.8594)

       Prediction
Actual   0   1
     0 738   9
     1 115  20

Validation Partition for Ordinal Logistic Model
Confusion Matrix (Accuracy 0.8418)

       Prediction
Actual   0   1
     0 480   6
     1  87  15
```

Nominal Logistic Model

a) Development and Analysis of Nominal Logistic Regression Model for Car Accidents Dataset

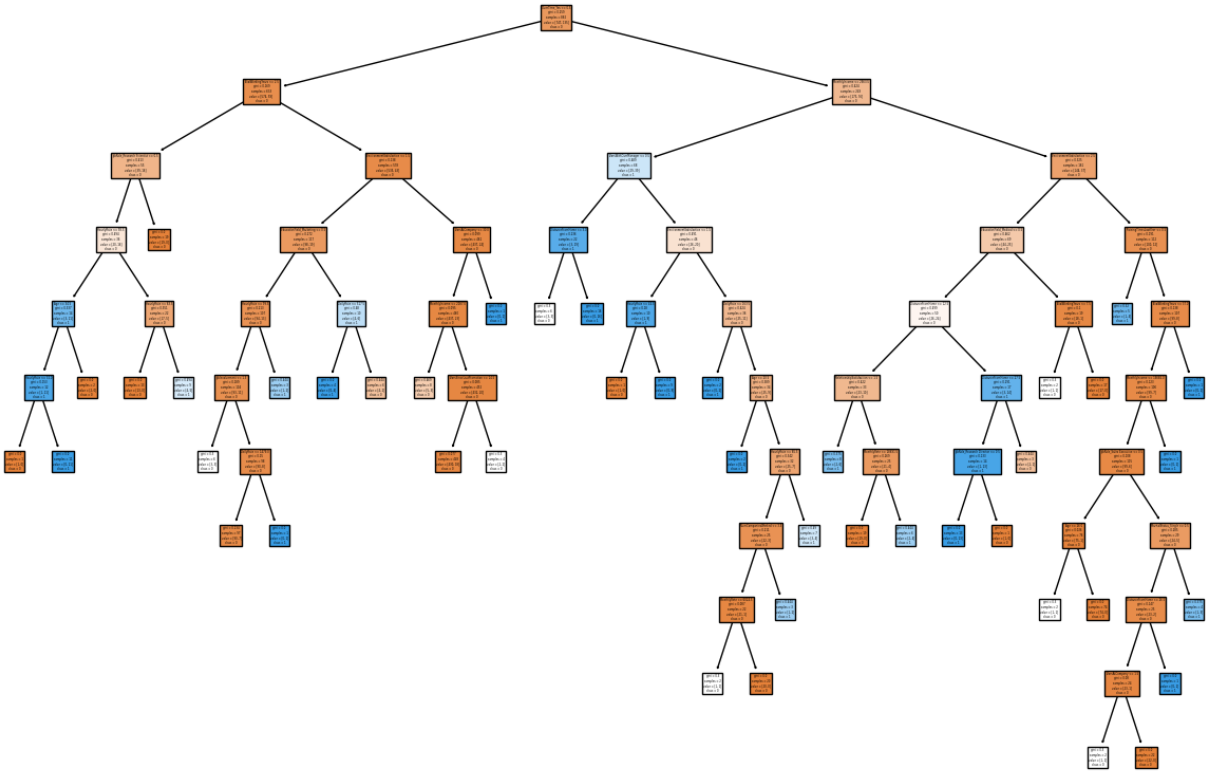
```
Nominal Logistic Regression
  Intercepts [0.089]
  Coefficients [[-0.013 -0.     0.023  0.138 -0.173  0.002 -0.189  0.021 -0.091 -0.
   0.     0.065 -0.016  0.152 -0.038 -0.237 -0.028 -0.042 -0.039  0.03
  -0.031  0.105 -0.086  0.309  0.04  -0.159  0.217 -0.035  0.203 -0.224
  -0.074  0.135  0.135  0.046  0.228 -0.    -0.101 -0.02  -0.125  0.001
   0.197 -0.082  0.327  0.796]]


Classification for First 10 Records in Validation Data Set
      Actual  Classification    P(0)    P(1)
1291       1               0  0.9717  0.0283
1153       1               1  0.2182  0.7818
720        1               1  0.4131  0.5869
763        0               0  0.6065  0.3935
976        0               0  0.8618  0.1382
724        0               0  0.9867  0.0133
314        0               0  0.9136  0.0864
258        0               0  0.9545  0.0455
442        0               0  0.7269  0.2731
1393       0               0  0.9415  0.0585
```

c) Confusion matrix for nominal Logistic Model

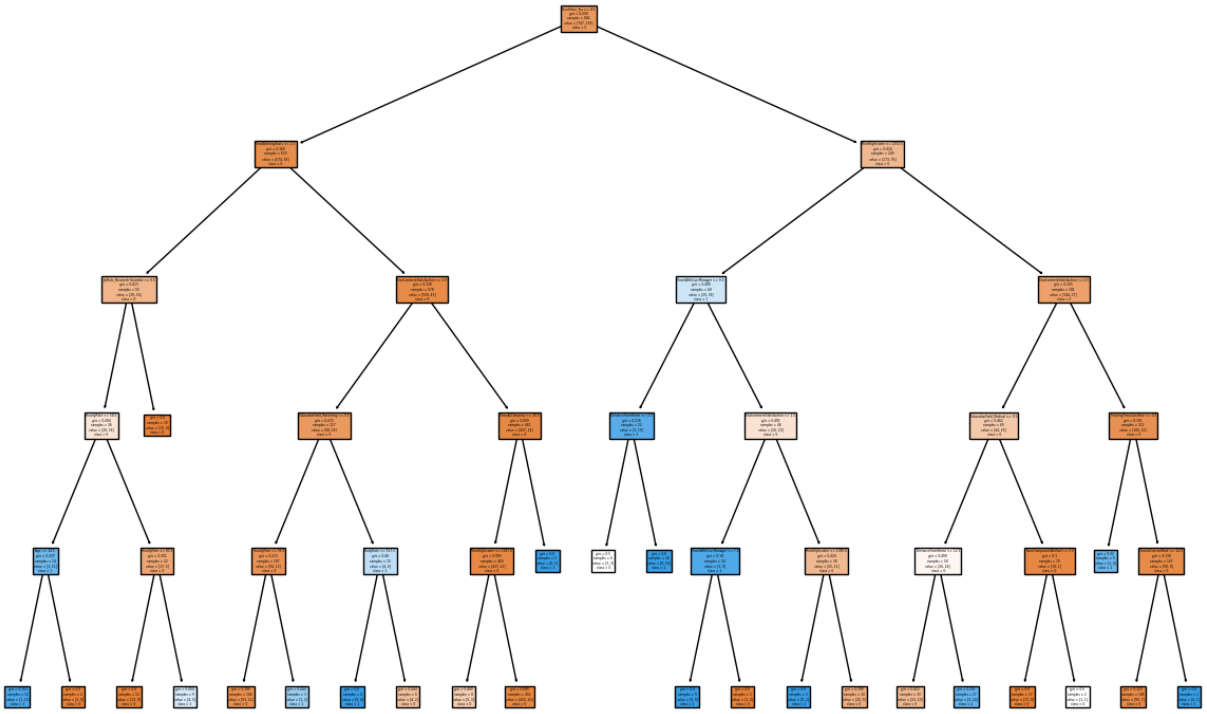
```
Training Partition for Nominal Logistic Model
Confusion Matrix (Accuracy 0.8889)

       Prediction
Actual   0   1
     0 731  16
     1  82  53

Validation Partition for Nominal Logistic Model
Confusion Matrix (Accuracy 0.8571)

       Prediction
Actual   0   1
     0 469  17
     1  67  35
```

Tree Classification:

a) **Visualization of Decision Tree Classifier for Classification**



Visualization of Decision Tree Classifier for Classification with MaxDepth = 5



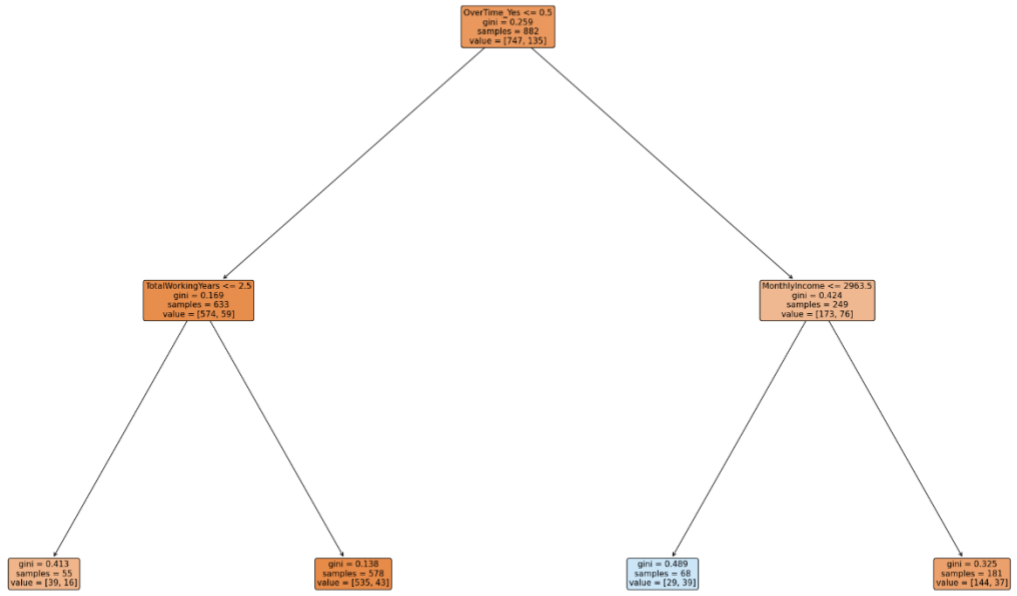
b) **Confusion matrix for Full Tree**

```
Training Partition for Full Tree
Confusion Matrix (Accuracy 0.9138)

       Prediction
Actual   0   1
     0 737  10
     1  66  69

Validation Partition for Full Tree
Confusion Matrix (Accuracy 0.8095)

       Prediction
Actual   0   1
     0 454  32
     1  80  22
```

c) Hyperparameter using Grid Search for Decision Tree Classifier

```
Improved score:0.8492
Improved parameters:  {'max_depth': 2, 'min_impurity_decrease': 0, 'min_samples_split': 5}
```

d)Best Classification Tree with GridSearch



d) Confusion Matrix for Best Classification Tree

```
Training Partition
Confusion Matrix (Accuracy 0.8583)

       Prediction
Actual   0   1
     0 718  29
     1  96  39

Validation Partition
Confusion Matrix (Accuracy 0.8384)

       Prediction
Actual   0   1
     0 468  18
     1  77  25
```

Using Neural Networks:

Training and Analysis of a Neural Network Model for Accidents Prediction

```
Final Intercepts for Accidents Neural Network Model
[array([ 47.36666264, -13.88488433,  34.35372855]), array([3.29468226])]

Network Weights for Accidents Neural Network Model
[array([[ 18.00691564,  -4.9330195 ,   4.25188861],
       [ 10.09486813, -11.10303496,  -0.6517032 ],
       [-14.13396475, -40.89449678,   7.95191889],
       [ 17.4591299 ,   2.16799605, -15.66237201],
       [ 15.39698307,   9.04472085,  14.3483274 ],
       [  3.96604029, -16.97636909,   3.5160607 ],
       [  6.46379413,  10.73162768,   6.09239173],
       [ -0.50041296,  -3.90814826,   1.01021971],
       [ 16.79782478,  15.01918181,  -8.28992153],
       [ 17.38650217,   0.31967725, -11.41476567],
       [ -4.72589925, -16.02918887,  23.46043618],
       [ -4.59445003, -41.96189402,   7.62885625],
       [ -7.80867368,  -6.95198097,  10.36310177],
       [ 14.47680191,   5.7078932 ,  -1.57536098],
       [  4.65935411,   9.49293034,   7.94882169],
       [  7.53619899,  16.82168084,   6.48849523],
       [-21.90201907,  19.8300012 ,  13.46435635],
       [ 13.78011596,  15.89090643,  -3.32745144],
       [  2.27680916,  -7.45653844,  17.05976471],
       [  7.37939238, -38.37766637,  15.83014189],
```

b) **Accident Severity Classification using Neural Network Model on Validation Set**

```
Classification for Accidents Data for Validation Partition
      Actual  Classification
1291       1               0
1153       1               1
720        1               1
763        0               1
976        0               0
724        0               0
314        0               0
258        0               0
442        0               0
1393       0               0
894        0               0
435        1               0
952        1               1
236        1               0
1170       0               0
1295       0               0
826        0               0
453        1               1
1230       0               0
702        0               0
1418       0               0
1252       0               0
299        0               0
```

c) **Confusion Matrix for Neural Network Model**

```
Training Partition for Neural Network Model
Confusion Matrix (Accuracy 0.9490)

       Prediction
Actual   0   1
     0 746   1
     1  44  91

Validation Partition for Neural Network Model
Confusion Matrix (Accuracy 0.8486)

       Prediction
Actual   0   1
     0 463  23
     1  66  36
```

d) **Identification of Optimal Hidden Layer Size using Grid Search**

```
Best score:0.8685
Best parameter:  {'hidden_layer_sizes': 2}
```

e) **Training and Analysis of Improved Neural Network Model for Accidents Prediction**

```
Final Intercepts for Accidents Neural Network Model
[array([ 65.22005005, -24.50462507]), array([-1.7664497])]

Network Weights for Accidents Neural Network Model
[array([[  7.50076353,   2.87359726],
       [ -5.48915158,  -3.97990728],
       [-16.25800076,  13.34733943],
       [  8.2340116 ,  12.90692315],
       [ 18.25296941,  -8.4332803 ],
       [ -3.20463667,  16.31503228],
       [ -1.24928995, -20.29623458],
       [-12.42940082,  13.32082548],
       [ 25.38572474,   4.93782947],
       [ 25.79368414, -13.97014661],
       [ 12.76933803,  25.78246849],
       [  3.65723389,  14.1001177 ],
       [ -4.81687911, -12.46947666],
       [  5.14362062, -15.75308431],
       [ 16.04001405,   3.83877492],
       [ 19.25628987, -16.51784668],
       [ 16.94671291,   7.93387853],
       [  3.73991398,  -6.55644385],
       [ 14.67222401,   4.2564862 ],
       [  7.91779872,  28.73534359],
```

e) Confusion Matrix for Neural Network Model for Accident Prediction

```
Training Partition for Neural Network Model
Confusion Matrix (Accuracy 0.9535)

       Prediction
Actual   0   1
     0 744   3
     1  38  97

Validation Partition for Neural Network Model
Confusion Matrix (Accuracy 0.8112)

       Prediction
Actual   0   1
     0 446  40
     1  71  31
```

In our approach, we considered multiple data mining techniques to handle the classification problem of predicting employee attrition. We evaluated Logistic Regression for its simplicity and interpretability, and Neural Networks for their ability to capture non-linear relationships. The choice of these techniques was driven by their different strengths in handling binary classification tasks and the nature of our dataset which includes both numeric and categorical variables.

Step 9 - Interpret the results of technique(s)/(models) and assess the technique(s) – compare them and select the best one

Model performance was evaluated using accuracy scores and confusion matrices. The Logistic Regression model provided a solid baseline with good interpretability of feature importance. The Neural Network, optimized through grid search, showed higher accuracy but at the cost of increased complexity. The decision on the final model took into account the trade-offs between accuracy and model simplicity.

Confusion matrices for multiple predictors logistic model

```
Training Partition
Confusion Matrix (Accuracy 0.9036)

       Prediction
Actual   0   1
     0 740   7
     1  78  57

Validation Partition
Confusion Matrix (Accuracy 0.8605)

       Prediction
Actual   0   1
     0 469  17
     1  65  37
```

Confusion matrix for training & validation partition

```
Training Partition for Ordinal Logistic Model
Confusion Matrix (Accuracy 0.8594)

       Prediction
Actual   0   1
     0 738   9
     1 115  20

Validation Partition for Ordinal Logistic Model
Confusion Matrix (Accuracy 0.8418)

       Prediction
Actual   0   1
     0 480   6
     1  87  15
```

CONCLUSION

The final step involves deploying the best-performing model into the operational system for real-time attrition prediction. This involves integrating the model into the existing HR infrastructure, ensuring scalability, reliability, and security. Continuous monitoring and evaluation of the deployed model are essential to adapt to changing dynamics and maintain optimal performance over time.

Comparative Analysis

1.The logistic regression models achieved the following accuracies:

- Logistic Model: Training Accuracy - 90.36%, Validation Partition Accuracy - 86.05%.
- Ordinal logistic Model: Training Accuracy - 85.94%, Validation Partition Accuracy - 84.18%.
- Nominal logistic Model: Training Accuracy - 88.89%, Validation Partition Accuracy - 85.71%.

2.Decision Trees:

- Controlled Decision Trees Model: Training Accuracy - 91.38%, Validation Partition Accuracy - 80.95%.
- Smallest tree: Training Accuracy - 85.83%, Validation Partition Accuracy - 83.84%.

Additionally, we developed a normalized neural network model with a Training Accuracy of 94.90% and Validation Partition Accuracy of 84.86%. Furthermore, we utilized GridSearch CV to optimize neural networks, achieving a Training Accuracy of 95.35% and Validation Partition Accuracy of 81.12%.

Training Accuracy: The Neural Network model shows the highest training accuracy at over 93%, indicating it's highly effective in fitting the training data. The Tree model follows with an accuracy of 91.38%.

Validation Accuracy: While the training accuracy for the Neural Network model is high, it doesn't perform as well on the validation set compared to the logistic models. The logistic model performs the best in the validation phase with an accuracy of 86.05%.

Overfitting: The Tree model shows signs of overfitting with a significant drop from training to validation accuracy (from 91.38% to 80.78%).

False Positives and Negatives: The Neural Network model demonstrates a strong ability to minimize false positives and negatives in the training set but not as effectively in the validation set.

Among these models, **the logistic regression model** emerged as the best-performing algorithm, achieving the highest validation partition accuracy rate of 86.05%. This model's simplicity, interpretability, and high predictive accuracy make it a valuable tool for organizations seeking to mitigate employee turnover and improve retention strategies.

These analyses suggest that while the Neural Network model excels in the training phase, Logistic regression might offer a better balance between training and validation performance, showing less propensity to overfit and maintaining decent accuracy on unseen data.

BIBLIOGRAPHY

- The dataset is from the Kaggle: https://www.kaggle.com/