1. Problem Description

Human resources has been using analytics for years. However, the collection, processing and analysis of data has been largely manual, and given the nature of human resources dynamics and HR KPIs, the approach has been constraining HR. The goal is to try to use predictive and descriptive analytics in identifying the employees most likely to get promoted.

1. Project Pipeline
   1. Load dataset and required libraries
   2. Basic dataset statistics + conclusions
      1. Shape of the dataset
      2. Viewing some rows from the dataset
      3. Description of the numerical attributes
      4. Description of the categorical attributes
   3. Checking for null values in columns
   4. Checking for columns with only one value
   5. Checking for duplicate rows
   6. Dropping the 'employee\_id' column as it's not important in the following steps
   7. Data Visualization
   8. Descriptive Analysis
   9. Data Preprocessing
   10. Training Models
2. Analysis and Solution of the problem
   1. Data preprocessing
      1. Handling categorical data using one hot encoding
      2. Handling unbalanced classes using SMOTE
      3. Splitting data into train-test-validation
      4. Normalize data values
   2. Data visualization
      1. Uni-variate:
         1. Boxplots (unsuccessful) 🡪 because features have different scales.
         2. Histogram distributions for numerical columns
         3. Bar Plots for categorical columns
         4. Bar plot for the Target column that shows there is a clear unbalancing in classes
      2. Multi-variate:
         1. Correlation heatmap between variables
         2. Crosstab plot between each feature and the target to show its effect on the target
   3. Extracting insights from data
      1. From statistical analysis:
         1. The distribution of data for features no\_of\_trainings, age, length\_of\_service, avg\_training\_score looks normal because it's clear that the mean and the median are close enough
         2. For categorical data:
            1. Data majority in gender is male with frequency 38496
            2. Data majority in department is Sales & Marketing with frequency 16840
            3. Data majority in education is Bachelor's with frequency 36669
            4. Data majority in region is region\_2 with frequency 12343
            5. Data majority in recruitment\_channel is other with frequency 30446
      2. From visualization:
         1. There is a clear unbalancing problem in the target column between the two classes
         2. From heatmap:
            1. Each variable has correlation = 1 with itself as expected
            2. There is high correlation between age and length\_of\_service as expected as older employees are more likely to have been working in the company for longer time
            3. KPIs\_met>80% is somehow related to previous year rating as expected as high rated employees are more likely to have their KPIs met the condition
         3. From crosstab:
            1. It's clear that the higher the avg\_training\_score, the more likely the employee is to be promoted
            2. The above graph shows that there is no biasedness over regions in terms of Promotion as all the regions share promotions almost equally
            3. It's clear that the more awards the employee have, the more likely he/she is to be promoted
            4. It's clear that the more KPIs the employee have, the more likely he/she is to be promoted
            5. It's clear that the higher the rating of the employee, the more likely he/she is to be promoted
            6. Overall, we don't see a pattern in the length of service to determine the promotion of the employee
            7. Overall, we don't see a pattern in the age to determine the promotion of the employee, and this shows that the company promotes employees of all ages equally, even the freshers have equal share of promotion.
            8. It's clear that the department has no pattern to determine the promotion of the employee, so, we can safely say that Departments have a similar effect on the promotion.
            9. The gender doesn't affect how likely the employee is to be promoted.
            10. It's clear that the less trainings the employee have, the more likely he/she is to be promoted
            11. It's clear that Education doesn't affect how likely the employee is to be promoted
            12. It's clear that recruitment\_channel doesn't affect how likely the employee is to be promoted.
      3. From descriptive analysis (association rules):
         1. because of the unbalancing in the data the insights are not good, but it make sense that when:
            1. when avg\_training\_score is low its more likely to be not promoted, with support= 0.210353 and confidence= 0.960510
            2. when KPIs\_met >80% is no its more likely to be not promoted, with support= 0.622373 and confidence= 0.960413
            3. when avg\_training\_score is high its more likely to be promoted, with support= 0.043589 and confidence= 0.142101
            4. when KPIs\_met >80% is yes its more likely to be promoted, with support= 0.059517 and confidence= 0.169094

And so on as we have deduce those results from previous visualization.

* 1. Model/classifier training:
     1. SVM
     2. KNN
     3. Naïve Bayes
     4. Logistic Regression
     5. Decision Trees
     6. Random Forest

1. Results and Evaluation:
   1. Without feature selection:

Using forward feature selection:

We conclude from results to drop gender and education.

* + 1. SVM:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.9 | 1.0 |
| recall | 1 | 0.89 |
| f1-score | 0.95 | 0.94 |
| accuracy | 0. 94509 | |

* + 1. KNN:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.93 | 0.93 |
| recall | 0.93 | 0.93 |
| f1-score | 0.93 | 0.93 |
| accuracy | 0. 92914 | |

* + 1. Naïve Bayes:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.87 | 0.68 |
| recall | 0.58 | 0.91 |
| f1-score | 0.69 | 0.78 |
| accuracy | 0.74409 | |

* + 1. Logistic Regression:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.91 | 0.98 |
| recall | 0.98 | 0.90 |
| f1-score | 0.94 | 0.94 |
| accuracy | 0.94104 | |

* + 1. Decision Trees:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.95 | 0.92 |
| recall | 0.92 | 0.95 |
| f1-score | 0.94 | 0.94 |
| accuracy | 0.936366 | |

* + 1. Random Forest:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.95 | 0.97 |
| recall | 0.97 | 0.95 |
| f1-score | 0.96 | 0.96 |
| accuracy | 0.95924 | |

* 1. With feature selection:
     1. SVM:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.89 | 0.99 |
| recall | 0.99 | 0.88 |
| f1-score | 0.94 | 0.93 |
| accuracy | 0.933998 | |

* + 1. KNN:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.93 | 0.94 |
| recall | 0.94 | 0.93 |
| f1-score | 0.94 | 0.93 |
| accuracy | 0.93487 | |

* + 1. Naïve Bayes:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.88 | 0.67 |
| recall | 0.56 | 0.92 |
| f1-score | 0.68 | 0.78 |
| accuracy | 0.738859 | |

* + 1. Logistic Regression:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.89 | 0.97 |
| Recall | 0.97 | 0.88 |
| f1-score | 0.93 | 0.92 |
| accuracy | 0.9239 | |

* + 1. Decision Trees:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.95 | 0.93 |
| recall | 0.93 | 0.95 |
| f1-score | 0.94 | 0.94 |
| accuracy | 0.93867 | |

* + 1. Random Forest:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| precision | 0.95 | 0.97 |
| recall | 0.97 | 0.95 |
| f1-score | 0.96 | 0.96 |
| accuracy | 0.96279 | |

We choose Random Forest as the final solution because:

* It has the highest accuracy.
* It make feature selection as it takes the majority vote of many decision trees each of them has different set of features (diversity).
* It is stable doesn’t get affected by noise.

1. Unsuccessful Trials

In model training:

* + 1. Naïve Bayes gives the worst accuracy

In association rules:

Because the data is unbalanced

1. Enhancements and Future Work
   1. More data visualization and analysis.
   2. Try out different feature selection method.
   3. Try out different learning models.
2. Map-Reduce :
   1. Technology used:
      1. Hadoop distribution system (pseudo distributed mode).
   2. Model used :
      1. KNN
   3. Methodology:
      1. Put the entire dataset in HDFS.
      2. Write one code file for mapper.
      3. Write another code file for reducer.
      4. Run Hadoop to predict the class of a given input.