Detecting Potential Candidates Who are Looking for New Job

# Problem Statement

How can a training institute expand their business to manpower recruitment by using the current credential/demographics/experience related data to predict the probability of an enrollee to look for a job change.

# Context

Company XYZ is a training institute which conducts training for analytics/ data science. They want to expand their business to manpower recruitment (data science only) by connecting their enrollees with their clients who are looking to hire employees working in the same domain. Before that, they want to know among the large number of signups, which of these candidates are looking for a new employment. To understand the factors that lead the enrollees to look for a job change, Company XYZ wants to build a model based on the current credentials/demographic/experience data they collected from the enrollee to predict the probability of them to look for a new job.

# Data Source

The datasets are sourced from <https://www.kaggle.com/aswathrao/hr-analysis> and consists of two datasets: train.csv, test.csv

* train.csv is composed of a total of 14 features collected from 18.4k enrollees, with 5 numerical features and 9 categorical features.
* test.csv is composed of a total of 13 features collected from 15k enrollees, with 4 numerical features and 9 categorical features.
* train.csv and test.csv are identical except there’s no outcome column in test.csv.

# Criteria For Success

* Determine the key features that lead the enrollee to look for new employment.
* Build a model based on the key features and predict the probability of the enrollee looking for a new job with at least 75% accuracy.

# Constraints & Scope

* Handling missing value in both train.csv and test.csv datasets. The missing value ratio can be as high as 30% in some features.
* Determine key features and build a model that detects a potential candidate who is open for new employment among all the enrollees.
* Adding salary information of current positions would be helpful but might be beyond the scope of this project.

# Approach

Multiple steps will be taken to build a predictive model for this project as well as to analyze the resulting predictions.

1. The train.csv and test.csv will be imported and cleaned via Python. Missing values will be handled appropriately based on specific factors.
2. Categorical variables will all be encoded to numerical variables using Ordinal Encoding, One Hot Encoding, or dummy variable encoding techniques, based on if the features have natural rank ordering or not.
3. The cleaned dataset will be explored visually in order to find interesting trends/correlations in the data. Pairs of columns with correlation coefficient higher than a threshold will be reduced to only one in order to avoid multicollinearity.
4. Multiple models including Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbors will be used to train the train.csv dataset after reducing multicollinearity from both the train.csv and test.csv.
5. Apply the trained models on the test.csv dataset, compare the accuracy of each model and pick up the one with the highest accuracy.

# Deliverables

The final draft of the project will be presented in the form of a slide deck and formal project report . Jupyter Notebooks will be delivered detailing each step taken and code written for the analysis of the project. A Github repository for the project will be created as well.