Project Dissertation Report On

ANALYSING AND PREDICTING JOB SEEKING DECISIONS OF STUDENTS TAKING PROFESSIONAL COURSES

Submitted By-

Name- Devanshi Mathur

Roll Numer-2K19/DMBA/026

Under the Guidance of-

Mr. Mohit Beniwal

Assistant Professor

DELHI SCHOOL OF MANAGEMENT

Delhi Technological University

Bawana Road, New Delhi, 110042

DECLARATION

I hereby declare that the Project Report entitled "Analysing and Predicting Job Seeking Decision In Students Taking Professional Courses" submitted to Delhi School of Management in partial fulfilment of the requirement for the award of the degree Master of Business Administration is a record of the research carried out by me under the guidance of Mr. Mohit Beniwal, Assistant Professor at Delhi School of Management, Delhi Technological University. I further declare that the work reported in this project has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Devanshi Mathur
2K19/DMBA/026

ACKNOWLEDGEMENTS

The completion of this study would have been impossible without the material and moral support from various people. It is my obligation therefore to extend my gratitude to them.

I am greatly indebted to Mr. Mohit Beniwal, who was my supervisor, for his effective supervision, dedication, availability and professional advice. I extend my gratitude to my lecturers who taught me in the MBA programme, therefore enriching my research with knowledge. The respondents deserve my appreciation for their support and willingness for providing the required information during my study. My appreciation finally goes to my classmates, with whom I weathered through the storms, giving each other encouragement and for their positive criticism.

EXECUTIVE SUMMARY

This study is conducted to analyse how some particular factors affect the job seeking decisions of candidates enrolled in enrolled in a professional training course and prepare a predictive model to predict whether a candidate who has enrolled in a training course is seeking employment with any firm or not. Primary data of 155 participants was collected through a structured questionnaire. Multiple models based on different classification algorithms were created and compared to find the most accurate one. The results indicated that the SMOTE Logistic Regression was most accurately predicting the class labels of the target variable, i.e., whether the person is 'job-seeker' or 'non-job seeker', with an accuracy score of 72.4%.

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Background

Job finding abilities in the job market depends on how much candidate skills are matching with job requirements. From a welfare standpoint, two aspects of ability matching are especially relevant. Firstly, the psychological benefit of being in a high-skill role is directly proportional to the worker's abilities. The psychological benefit of role difficulty can be zero or even below that at low levels of worker ability. Second, the payoff to worker skills (such as education) is positively related to the job's skillsets, both financially and psychologically. There is also a widely held but debatable belief that overall levels of required job skills have increased very fast, owing primarily to 2 advancements - technological changes and globalisation.

Professional certificate courses are a collection of in-demand courses designed to help people develop or improve essential skills for a particular job. These courses, developed by business leaders, leading universities or training agencies, help improve the skills and expertise required for best jobs currently in the industry through a versatile and accessible learning process. The students and working professionals all around the globe are also going for professional trainings apart from their regular full-time degrees in order to widen their skillset and prepare themselves better for their working world. Such training programs provide them with competitive edge in the job market and help them in jumping to the next levels in their careers. These training programmes are designed for specific roles and career pathways, providing skills-based preparation in particular areas where employers are looking only for best players. Job seekers feel assured that after completing such courses, they will have acquired the provable skills they need to make a significant impact on a company. Employers will be able

to see that they have this particularly required skill set. They can also display these qualifications by adding their certificates on a resume or LinkedIn profile. It'll also help them stand out from the rest. A few of the reason why students enrol in certification courses are as follows:

- To gain knowledge of a trade.

A certification course is much more demanded than a bachelor's or associate's degree for a lot of professions.

- To become an expert in a particular area.

Professionals with university degrees in areas like nursing or psychology might sometimes have to enrol in a training program to broaden their knowledge or learn skills.

- To change jobs.

When someone want to switch their career, a certificate course can be a good place to start.

- As a backup strategy.

If someone’s present job is not helping them to sustain themselves, a good professional training can provide them with real skills that can be converted to a better paying career.

- To improve chances of landing a job.

When applying for jobs in today's competitive job market, people may find themselves competing against hundreds of other applicants. A good certification help them stand out.

Even as the pandemic caused chaos in the job market, a lot of industries stopped their hiring processes. College students were unable to find good jobs and many industry professionals were laid off by the companies. So, a lot of them planned to enrol in full-time or part-times

education or training programs for a specific career or trade focusing on practical applications. According to a study released by an assistant professor of psychology at Penn State University, job seekers who participate in career service training programs are 2.67 times more likely to find employment than those who do not participate. They also found these programs to be effective for both younger job seekers as well as those over 50 years old. There are a lot of training institutes in India that provide training in exciting fields, like Digital Marketing, Virtual Reality, Web Development, Artificial Intelligence, and hundreds of other fields.

1.2 Problem Statement

Organisations also want to hire candidates who have successfully completed such professional courses provided by these training agencies. Training agencies tie up with various firms and connect the candidates enrolled in their courses to the firms who are also hiring for the same domain. Though a lot of candidates signup for the training, these training organisations or agencies usually want to know if candidates want to work with any firm after training or they are not seeking any employment. Knowing the probability of this beforehand becomes really helpful for the agencies as it helps them know how many students are expecting to be placed from their institute. And thus, it helps them to organise their placements accordingly. It also helps them to plan the courses accordingly and the most important of all, categorize the candidates and plan the placement process accordingly.

1.3 Objective

This project aims to analyse some particular factors affect the job seeking decisions of candidates enrolled in a professional training course and create a predictive model of whether a candidate is seeking employment with a firm or not based on the information regarding

candidate demographics, relevant experience, education level, company type and size, training hours etc, using various machine learning classification algorithms and visualisations.

1.4 Scope

The project and the predictive model prepared in this project is intended to apply only to people who have enrolled in some training courses. It is not applicable to any student or any industry professional who is not pursuing any kind of vocational training.

2. LITERATURE REVIEW

There has been only little research done in the area related to job seeking patterns and factors affecting it.

(Turban & Keon, 1993) found that the work responsibilities, the organisation itself, prospects for promotion, future relationships with co-workers, compensation, and job security are the most important considerations for an applicant to accept a job.

(Barber and Roehling, 1993) Some relevant researches have also found that remuneration, the ability to apply expertise and skills, and demanding and exciting work all play a role in an individual's decision to apply for a job.

(Khan, Awang and Ghouri, 2013) The findings of the study showed that the effectiveness of recruitment depends upon the placement of advertisement and salary is the most influential motivator to find interest in the job applied. The study also found that the recruitment sources and applicant’s perceptions of job significantly influence the intention to pursue the position applied by the job seeker.

(Wanberg, Kanfer and Rotundo, 1999) In this analysis by 3 individual differences variables - job-search motivations, competencies, and constraints were investigated as predictor variables of job-search intensity. They also examined the longitudinal relationship between intensity of the job-search and reemployment success. Results show that there is a strong relationship between the predictor variables motivation control, self-efficacy, job-search, financial hardship and employment commitment, and the predicted variable job-search intensity. Any connection between perceived job-search constraints and job-search intensity was not discovered. For

those who have been continuously unemployed, motivation control was identified as the only lagged predictor of job-search intensity over time.

Hooft and Jong (2009) studied the sample of 138 temporary workers in The Netherlands and found that the most strongly linked behavioural beliefs to intentions to pursue temporary employment were sense of security, work–life balance, and status. People with a low level of collectivism were also found to be more driven by their personal attitude toward job seeking and less by societal pressures than those with a high level of collectivism.

(Gatewood, Gowan and Lautenschlager, 1993) In this study, different aspects of corporate image and recruitment image were examined. It was found that that organisation's image is linked to the data information available about it. Also, different applicants might have different and recruitment and corporate images of the same companies, and both these images are strong determinants of initial decisions about making any professional links with the organisation.

(Catanzaro, Moore and Marshall, 2010) The aim of this study was to see how attitudes about the organization's culture influence male and female job search and application decisions. The findings suggest that organisational culture and gender combine to affect applicant interest. Males are more likely than females to choose to work for a competitive company; however, the majority of both males and females expressed a greater desire to work for a positive company, despite the lower pay.

According to the United States Department of Education's 2004 National Assessment of Vocational Education, close to half among all high-school students enrolled in some kind of vocational training that year, even if it was only single program. The study discovered that vocational training in high-school had a positive impact on short and medium term earnings.

Hooft, Kammeyer-Mueller, Wanberg and Kanfer, 2020) To test relationships between job-search self-regulation, job-search activity, and employment performance outcomes, this study conducted a quantitative literature review. key antecedents of job-search self-regulation, job-search attitudes, and work performance quantitatively (i.e., personality, attitudinal influences, and contextual variables) were also examined. Overall, the intensity of job search did not predict the quality of the job. Job-search self-control and job-search quality were found as promising frameworks to be researched upon in future, as both and employment quality and quantitative employment success outcomes were predicted.

(Bolliger, 2004) This research found that teacher variables, technical problems, and interactivity are the three constructs that affect student satisfaction with online courses. The Online Course Satisfaction Survey was completed by 155 people out of a total of 303 online students.

(Nonis and Fenner, 2021) This study of students taking online classes revealed key student expectations that influence their decision to take online/web-assisted courses. It was found that convenience, enjoyment & independence are two factors that affect the motivations for students taking online/web-assisted courses

2.1 Research Gap

Although there has been a lot of research done in the area of what factors affect the decision of a candidate to apply for a particular job or not and what are the expectations of students from various training and online courses, there has been no significant research done in the area related to the job seeking patterns in people enrolled in training various training courses. There

has no research in particular related to whether a student in a professional training course is seeking job or not after completing the course. This study will be tapping this research gap to analyse how some particular factors affect the job seeking decisions of candidates enrolled in the course and to predict whether a candidate is seeking job or not by applying various machine learning algorithms and comparing them to get the most accurate results.

3. DATA COLLECTION AND METHODOLOGY

3.1 Data Collection and Metadata

The data used in this study is the primary data. A structured questionnaire was created using Google Forms and was sent to different people in various universities, schools and working at companies who are taking some kind of extra course at any training institute. A total of 155 responses were received. Both descriptive and predictive analysis of the data has been done.

In this study, the job-seeking decision of the students is measured by 9 variables: gender of the candidate, types of enrolment in the training institute, candidate's maximum education level, work experience of the candidate, size of the company that they work in, type of company that they work in, last time that they changed their jobs and the number of training hours. The target variable is whether they are seeking job or not. The Job seekers are valued as 1 and non-job seekers are valued as 0.

3.2 Methodology

The data collected will be analysed using Descriptive Data Analysis and Predictive Data Analysis. Descriptive Data Analysis wil be focusing on exploring the different aspects of the dataset using graphs and treemaps. Descriptive Analysis has been done using data visualisation tool by Google – Google Studio. The probability weather the candidate is seeking a new job or not is predicted using Predictive Data Analysis. This section will be comparing multiple models created using different Machine Learning Algorithms. The ML algorithms that will be used in the Predictve Data Analysis are Support Vector Machine, Decision Tree, Random Forest,

Tuned Random Forest, Logistic Regression and K-Nearest Neighbour. Then two of these models having best results will be further improved using SMOTE. SMOTE is an oversampling technique used to deal with imbalanced datasets. The aim will be to select the best model with highest Accuracy and ROC AUC Score. The tool used for the data analysis is Python. Major Python libraries used in the project are numpy, pandas, matplotlib, pyplot, gridspec, seaborn, squarify and sklearn.

4. THEORITICAL BACKGROUND

Following is the theoritical explanation of the algorithms compared in this projet so as to reach most accurate model.

4.1 Support Vector Machine

Support Vector Machine (SVM) is a supervised machine learning lgorithm which is used for classification purposes. This algorithm aims to identify a hyperplane in an n-dimensional (n is the number of features/columns) space to distinctly classify the data points on it. Each datapoint is plotted as a point in the n-dimensional space. The value of each feature is the value of a particular coordinate. It then classify the datasets by identifying a hyperplane that clearly differntiates the two classes.

4.2 Decision Tree Algorithm

Decision Tree is also a supervised learning algorithm used for solving classification anf regression problems. The training model created by this algorithm predicts the class of the target variable by learning simple decision rules that have been inferred from the train data. Decision trees have root nodes that represents the entire population and are further divided into two or more homogenous sets., decision nodes, which is splitted into further subnodes and leaf nodes denotes the classifiers. Each node apart from the leaf nodes acts as a test case and each branch going down from that node provides are answers to that test case. The algorithm starts at the root of the tree and compares the root attributes values with the record's attribute value. On this basis, it follows the branch that correspond to that value and jump to the next node. This keeps repeating for every subtree rooted at a node until it reaches the leaf node.

Once reached the leaf node, the data is classified and can be said to be belonging to that particularclass.

4.3 Random Forest Algorithm

Random Forest, which is nothing but a collection of decision trees, is also a supervised learning algorithm used for regression and classification. In this algorithm, multiple decision trees are trained by randomly selecting differen subsets of data and bagged together to get more stable and accurate results. These subsets are made by randomly selecting x features and y observations from the original dataset. The classification of an entity is done by slecting the mode of classes predicted by all the individual decision trees. Decision Trees sometimes tend to overfit their training dataset. Random Forest correct this behaviour of overfitting.

4.4 Tuned Random Forest

There are various hyper-parameters in Random Forest that can be tuned to either improve the prediction of the model or to make training the model easier. Some of the parameters that we can play with to improve our model development process are:

- n_estimators: This is the numbers of decision trees to be trained and made. The algorithm is less likely to overfit if the number of trees will be increased. Therefore, this parameter can be increased to make a better model.
- max_features: This number should be reduced or kept balanced. It defines the number of features each tree is randomly assigned.
- min_sample_leaf: Model with a small leaf size is more prone to capture noise in the data. Therefore, multiple leaf sizes should be tried to find the most optimum for the case.

4.5 Logistic Regression

Logistic Regression is a supervised predictive modelling technique generally used for classificaation problems. It is used when the dependent variable has two levels, i.e. the outcome is binary. The dependednt variable is usually dichotomous or categorical in nature, i.e, it fits into one of two categories – 0 or 1, yes or no, pass or fail etc.

4.6 K-Nearest Neighbour

The KNN algorithm is a basic supervised machine learning algorithm that can be used to solve classification and regression problems. This algorithm searches for all the training instances that have attributes that are identical to the test example's attributes. These instances, referred to as ‘nearest neighbours’, may be used to evaluate the test example's class. The following statement best exemplifies the rationale for using nearest neighbours: "If it walks like a duck, quacks like a duck, and looks like a duck, then it's probably a duck." Each instace is represented by a data point in a d-dimensional space by the nearest neighbour classifier, where d is the number of features. Using one of the proximity indicators, it calculate the distance between a test instance and the rest of the data points in the training set. The k-nearest neighbours of a given instance x are the k points which are nearest to x.

4.7 Comparison Metrics

These models are compared using various metrics used in classification problems. Metrics used in the project for comparing the performance of the models are-

Accuracy

Accuracy in classification is defined as the ratio of the number of correct predictions made to

total number of predictions.

Classification accuracy = Correct predictions / Total predictions

Or

Accuracy = True Positive + True Negative / True Positive +True Negative + False Positive + False Negative

Recall

Recall in classification is the the ratio of correct positive predictions to the total positive examples in the dataset i.e. including those which were falsely predicted as negative but were in fact positive. It is also referred to as sensitivity.

Recall = True Positive / True Positive + False Negative

The Recall Score signifies the ability of the model to find all the positive examples., i.e to correctly classify total relevant results .

Precision

Precision in classification is the the ratio of correct positive predictions to the total positives predicted by the model. i.e. even including the false positives.

Precision = True Positive / True Positive + False Positive.

The Precision Score signifies how much percentage of the results is actually relevant.

F-Score

F-Score, also called F1 score, is a measure of a model's accuracy on a dataset. F-score combines the precision and recall of the model by taking their harmonic mean.

F Score = 2 * (Precision * Recall) / (Precision + Recall)

F score has its best value at 1 and worst at 0. The relative contribution of precision and recall to the F score are equal.

ROC-AUC Score

An ROC curve (Receiver Operating Characteristic Curve) is an evaluation metric for measuring the performance of binary classification problems. ROC is a probability curve in which True Positive Rate / Recall / Sensitivity is plotted against False Positive Rate / 1-Specificity. AUC (Area Under Curve) represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. A high AUC-ROC Score means the model is good at predicting 0s as 0s and 1s as 1s.

5. DESCRIPTIVE ANALYSIS

1- Job Seekers 0- Non-Job Seekers

5.1 Balanced Or Imbalanced Dataset

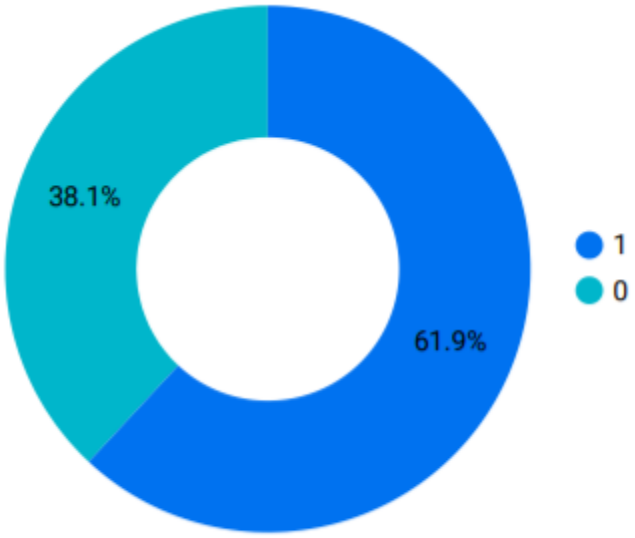

Figure 1- Imbalanced Data Set

As can be seen from the above figure, our dataset contains majority of non-job-seekers. The percentage of job seekers is 62% and that of non-job-seekers is 38%. We can say that our dataset is imbalanced. In an imbalanced classification, the distribution of the classes across the dataset is uneven. Although the dataset is only slightly imbalanced with a ratio of approximately 6:4, and will not be a concern for our predictive modelling. However, we will be dealing with the imbalance using SMOTE while doing the predictive analysis.

5.2 Gender-Based Analysis

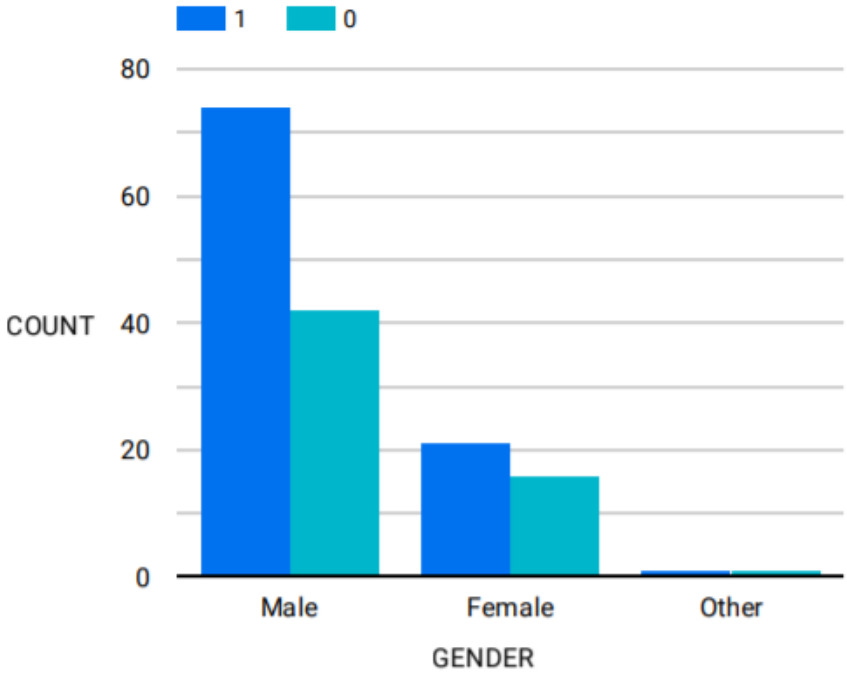

Figure 2 - Gender Based Analysis

From the above graph, we can easily infer that most job-seekers as well as non-job-seekers are male. This is due to the fact that our data is composed of more males than females.

5.3 Analysis Based On Education Level And Last Job Change

In the past one year, job-seekers have changed their jobs more than non-job seekers. Most of those who have never worked or have never searched for a job are also seeking jobs this year. Graduated students are more actively seeking jobs as compared to post-graduated or PhD candidates. Small amount of motivation can also be seen in school students in searching for jobs to become more independent.

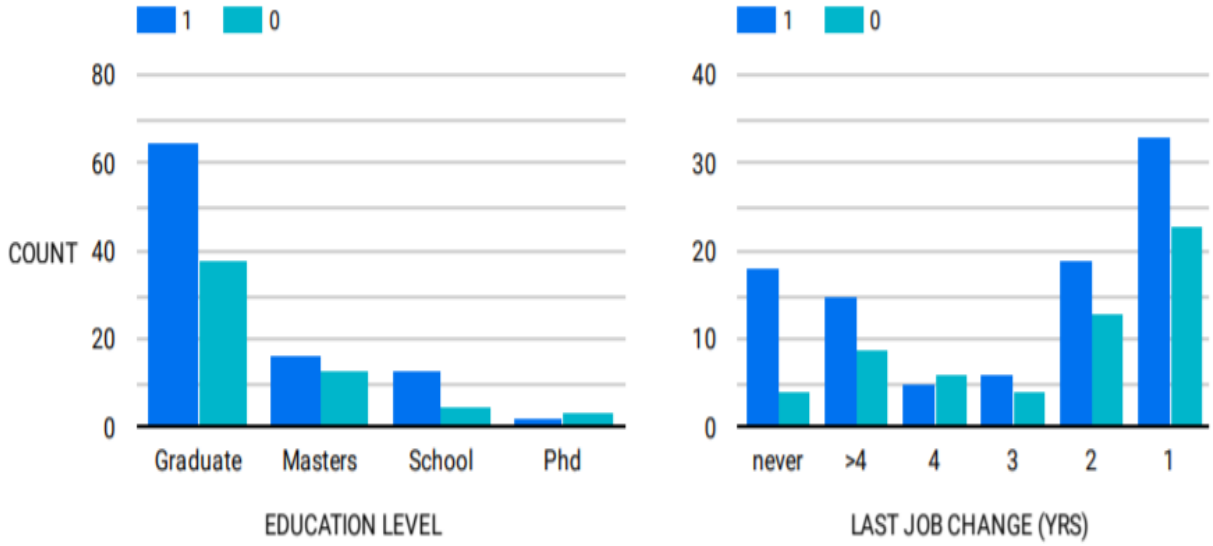

Figure 3 - Analysis Based on Education Level and Last Job Change

5.4 Analysis Based on Company Size and Employee Experience

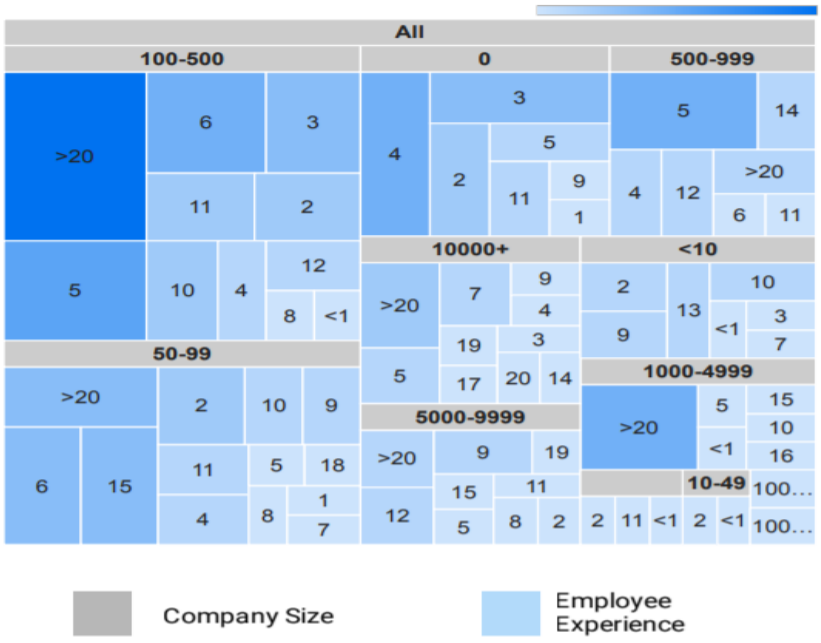

Figure 4 - Heatmap between Employee Experience and Company Size

By analysing the above tree map, we can infer that people with 20+ years of experience are dominating the workforce at all companies irrespective of the company sizes. We also observe that they are not seeking any new roles as they might be at their desired seniority level with the desired pay. They just wanted to upskill themselves and learn recent technologies like big data and machine learning. However, those who are in very small sized companies are still looking for a job change as they would want to settle in a big established corporation with a good brand name and gaining skills through training would help them do so.

6. PREDICTIVE ANALYSIS

6.1 Data Preparation and Code Generation

For predictive analysis, the dataset was first divided into training and test data using *train_test_split*() function from *sklearn.model_selection* package in Python. The train and test dataset have been divided into ratio of 6:4. That is, the model is trained using training dataset which consists of 60% data and then tested on the test dataset consisting of 40% entries.

The training set has been trained on different supervised and un-supervised machine learning classification models. The results have been compared along-side each other. The theoritical aspects of the algorithms used are discussed in the Theoritical Background section. Following are the different algorithms, Python libraries and functions used in the analysis, and their corresponding outputs –

6.1.1 Support Vector Machine

Package- *sklearn.svm, sklearn.pipeline*

Library- *Pipeline, svm, SVC*

Function- *make_pipeline(StandardScaler(), SVC(kernel='sigmoid'))*

```
# Support Vector Machine (SVM / SVC)

pipeline = make_pipeline(StandardScaler(), SVC(kernel='sigmoid'))
pipeline.fit(X_train, y_train)
svc_prediction = pipeline.predict(X_test)
cm_svc = confusion_matrix(y_test, svc_prediction)
svc_df = pd.DataFrame(data=[accuracy_score(y_test, svc_prediction), recall_score(y_test, svc_prediction),
                    precision_score(y_test, svc_prediction), roc_auc_score(y_test, svc_prediction),
                             f1_score(y_test, svc_prediction)],
             columns=['SVC Score'],
             index=["Accuracy", "Recall", "Precision", "ROC AUC Score","f1 Score"])
print(cm_svc)
print(svc_df)
```

```
[[ 2 25]
 [ 2 29]]
              SVC Score
Accuracy       0.534483
Recall         0.935484
Precision      0.537037
ROC AUC Score  0.504779
f1 Score       0.682353
```

Figure 5 - SVC Code

6.1.2 Decision Tree Algorithm

Package- *sklearn.tree*

Library- *DecisionTreeClassifier*

```
# Basic Decision Tree
dtree = DecisionTreeClassifier()
dtree.fit(X_train,y_train)
dtree_prediction = dtree.predict(X_test)
cm_dtree = confusion_matrix(y_test, dtree_prediction)
dtree_df = pd.DataFrame(data=[accuracy_score(y_test, dtree_prediction), recall_score(y_test, dtree_prediction),
                    precision_score(y_test, dtree_prediction), roc_auc_score(y_test, dtree_prediction),
                              f1_score(y_test, dtree_prediction)],
             columns=['Decision Tree Score'],
             index=["Accuracy", "Recall", "Precision", "ROC AUC Score", "f1 score"])
print(cm_dtree)
print(dtree_df)
```

```
[[ 9 18]
 [ 7 24]]
              Decision Tree Score
Accuracy                 0.568966
Recall                   0.774194
Precision                0.571429
ROC AUC Score            0.553763
f1 score                 0.657534
```

Figure 6 - Decision Tree Code

6.1.3 Random Forest Algorithm

Package- *sklearn.ensemble*

Library- *RandomForestClassifier*

```
# Random Forest
rfc = RandomForestClassifier(n_estimators=600)
rfc.fit(X_train,y_train)
rfc_prediction = rfc.predict(X_test)
cm_rfc = confusion_matrix(y_test, rfc_prediction)
rfc_df = pd.DataFrame(data=[accuracy_score(y_test, rfc_prediction), recall_score(y_test, rfc_prediction),
                  precision_score(y_test, rfc_prediction), roc_auc_score(y_test, rfc_prediction),
                           f1_score(y_test, rfc_prediction)],
             columns=['Random Forest Score'],
             index=["Accuracy", "Recall", "Precision", "ROC AUC Score", "f1 Score"])
print(cm_rfc)
print(rfc_df)
```

```
[[ 6 21]
 [ 3 28]]
              Random Forest Score
Accuracy                 0.586207
Recall                   0.903226
Precision                0.571429
ROC AUC Score            0.562724
f1 Score                 0.700000
```

Figure 7 - Random Forest Code

6.1.4 Tuned Random Forest

Package- *skearn.ensemble*

Library- *RandomForestClassifier*

```
# Tuned Random Forest
rfc1=RandomForestClassifier(random_state=0, n_estimators= 800, criterion = 'gini',max_features = 'auto',max_depth = 8)
rfc1.fit(X_train,y_train)
prediction_rf1= rfc1.predict(X_test)
cm_trfc = confusion_matrix(y_test, prediction_rf1)
trfc_df = pd.DataFrame(data=[accuracy_score(y_test, prediction_rf1), recall_score(y_test, prediction_rf1),
                  precision_score(y_test, prediction_rf1), roc_auc_score(y_test, prediction_rf1),
                           f1_score(y_test, prediction_rf1)],
             columns=['Tuned Random Forest Score'],
             index=["Accuracy", "Recall", "Precision", "ROC AUC Score", "f1 Score"])
print(cm_trfc)
print(trfc_df)
```

```
[[ 4 23]
 [ 3 28]]
              Tuned Random Forest Score
Accuracy                       0.551724
Recall                         0.903226
Precision                      0.549020
ROC AUC Score                  0.525687
f1 Score                       0.682927
```

Figure 8 - Tuned Random Forest Code

6.1.5 Logistic Regression

Package- *sklearn.linear_model*

Library- *LogisticRegression*

```
#Logistic Regression


logmodel = LogisticRegression()
logmodel.fit(X_train,y_train)
log_prediction = logmodel.predict(X_test)
cm_log = confusion_matrix(y_test, log_prediction)
log_df = pd.DataFrame(data=[accuracy_score(y_test, log_prediction), recall_score(y_test, log_prediction),
                  precision_score(y_test, log_prediction), roc_auc_score(y_test, log_prediction),
                           f1_score(y_test, log_prediction)],
             columns=['Logisitc Regression Score'],
             index=["Accuracy", "Recall", "Precision", "ROC AUC Score", "f1 score"])
print(cm_log)
print(log_df)
```

```
[[ 6 21]
 [ 5 26]]
              Logisitc Regression Score
Accuracy                       0.551724
Recall                         0.838710
Precision                      0.553191
ROC AUC Score                  0.530466
f1 score                       0.666667
```

Figure 9 - Logistic Regression Code

6.1.6 K-Nearest Neighbour

Package- *sklearn.neighbors*

Library- *KNeighborsClassifier*

```
# K-Nearest Neighbours (KNN)
# searched already to find optimal neighbours, removed from notebook as took a long time
knn = KNeighborsClassifier(n_neighbors=17)
knn.fit(X_train,y_train)
knn_prediction = knn.predict(X_test)
cm_knn = confusion_matrix(y_test, knn_prediction)
knn_df = pd.DataFrame(data=[accuracy_score(y_test, knn_prediction), recall_score(y_test, knn_prediction),
                  precision_score(y_test, knn_prediction), roc_auc_score(y_test, knn_prediction),
                          f1_score(y_test, knn_prediction)],
             columns=['KNN Score'],
             index=["Accuracy", "Recall", "Precision", "ROC AUC Score", "f1 score"])
print(cm_knn)
print(knn_df)
```

```
[[ 0 27]
 [ 0 31]]
               KNN Score
Accuracy        0.534483
Recall          1.000000
Precision       0.534483
ROC AUC Score   0.500000
f1 score        0.696629
```

Figure 10 - KNN Code

The model performance evaluation metrics used are accuracy, recall, precision, f1 score and ROC-AUC Score. The libraries used for these evaluation metrics are *accuracy_score, recall_score, precision_score roc_auc_score and f1_score*. These libraries have been imported from *sklearn.metrics*.

6.2 Summary and Analysis

The above results have been summarised in the table below.

Table 1 - Comparison of the Results

Algorithm	Accuracy	Recall	Precision	F1 Score	ROC-AUC Score	Comments
Support Vector	53.4%	93.5%	53.7%	68.2%	50.5%	Not able to distinguish
Decision Tree	56.9%	77.4%	57.1%	65.8%	55.4%	Not able to distingush
Random Forest	58.6%	90.3%	57.1%	70.0%	56.3%	Not able to distinguish
Tuned Random Forest	55.2%	90.3%	54.9%	68.3%	52.6%	Not able to distinguish
Logistic Regression	55.2%	83.9%	55.3%	66.7%	53.0%	Not able to distinguish
K-Nearest Neighbour	53.4%	100.0%	53.4%	69.7%	50.0%	Not able to distinguish

From the Table 1, it can be observed that both accuracy for all the algorithms is coming out to be around 50% and recall is coming good for all of them. But if we look at the ROC-AUC Score, we can see that all the models have ROC-AUC Score of about 50-55%. This means that

the models are not able to distingush between the two classes, i.e. job-seekers and non job-seekers.

This is happening because the data is suffering from imbalance, i.e. the number of examples of one class is much less than that of the other class. Due to this, the classification models were not able to effectively learn the decision boundary. To overcome this issue, generally, the examples in the minority class are oversampled by duplicating them in the training dataset before fitting the predictive models. But rather than duplicating, new examples of the minority class can be synthesized which is more effective than simply duplicating the minority class examples.

SMOTE-

In this project, Synthetic Minority Oversampling Technique (SMOTE) is used. In this technique, first we choose a random example A from the minority class and then find k nearest neighbours for that example. Then we pick up a random neighbour from the selected ones B and create a synthetic example at a randomly slected point between the two examples in the feature space. The newly created synthetic examples are a convex combination of the two chosen examples A and B. This process can be used to create as many as required synthetic examples. Following is the algorithm and output after performing SMOTE technique.

```
# Our data is biased, we can fix this with SMOTE

from imblearn.over_sampling import SMOTE

X = aug_train.dropna().drop(columns=['target']).values
y = aug_train.dropna()['target'].values

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

oversample = SMOTE()
X_train_resh, y_train_resh = oversample.fit_resample(X_train, y_train.ravel())
```

```
# Scale our data in pipeline

rf_pipeline = Pipeline(steps = [('scale',StandardScaler()),('RF',RandomForestClassifier())])
logreg_pipeline = Pipeline(steps = [('scale',StandardScaler()),('LR',LogisticRegression())])
```

Figure 11 - SMOTE Code (Part 1)

```
from sklearn.metrics import accuracy_score, recall_score, roc_auc_score, precision_score, f1_score

rf_pipeline.fit(X_train_resh,y_train_resh)
logreg_pipeline.fit(X_train_resh,y_train_resh)

rf_pred   =rf_pipeline.predict(X_test)
logreg_pred   = logreg_pipeline.predict(X_test)

rf_cm  = confusion_matrix(y_test,rf_pred )
logreg_cm  = confusion_matrix(y_test,logreg_pred )

rf_f1  = f1_score(y_test,rf_pred)
logreg_f1  = f1_score(y_test,logreg_pred)
```

Figure 12 - SMOTE Code (Part 2)

```
smote_rf_df = pd.DataFrame(data=[accuracy_score(y_test, rf_pred), recall_score(y_test, rf_pred),
                 precision_score(y_test, rf_pred), roc_auc_score(y_test, rf_pred), f1_score(y_test, rf_pred)],
             columns=['SMOTE Random Forest Score'],
             index=["Accuracy", "Recall", "Precision", "ROC AUC Score", "f1 Score"])

smote_logreg_df = pd.DataFrame(data=[accuracy_score(y_test, logreg_pred), recall_score(y_test, logreg_pred),
                 precision_score(y_test, logreg_pred), roc_auc_score(y_test, logreg_pred), f1_score(y_test, logreg_pred)],
             columns=['SMOTE Logistic Regression Score'],
             index=["Accuracy", "Recall", "Precision", "ROC AUC Score","f1 Score"])


smote_cm_log = confusion_matrix(y_test, logreg_pred)
smote_cm_rf = confusion_matrix(y_test, rf_pred)

print(smote_cm_rf)
print(smote_rf_df)
print(smote_cm_log)
print(smote_logreg_df)
```

```
[[ 6  5]
 [ 7 11]]
           SMOTE Random Forest Score
Accuracy                    0.586207
Recall                      0.611111
Precision                   0.687500
ROC AUC Score               0.578283
f1 Score                    0.647059
[[ 7  4]
 [ 4 14]]
           SMOTE Logistic Regression Score
Accuracy                          0.724138
Recall                            0.777778
Precision                         0.777778
ROC AUC Score                     0.707071
f1 Score                          0.777778
```

Figure 13 - SMOTE Code (Part 3)

It can be observed from table 2 that after applying SMOTE Technique, the Logistic Regression algorithm performed much better than Random Forest. The ROC-AUC Score for SMOTE Logistic Regression came out to be highest at 70.7%. It outperformed in terms of accuracy as well with 72.4% accurate predictions, without sacrificing its performance in other metrics, i.e recall and precision with the scores 77.8% for both. Overall, we can say that SMOTE +

Table 2 - Comparison of SMOTE Results

Algorithm	Accuracy	Recall	Precision	ROC-AUC Score	Comments
SMOTE Random Forest	58.6%	61.1%	68.7%	57.8%	Not able to distinguish
SMOTE Logistic Regression	72.4%	77.8%	77.8%	70.7%	Easily able to distinguish

Logistic Regression Model is the best among all the models that we analysed. And this model can certainly be used for predicting job seeking decisions of the candidates at training institues in future.

7. CONCLUSION

The study revealed that Logistic Regression along with SMOTE Technique together is able to most accurately predict whether a candidate who has enrolled in a training institute is seeking employment or not. The accuracy score for SMOTE Logistic Regression came out to be highest which was 72.4%, along with good Recall and Precision, both being 77.8%. The ROC-AUC Score also came out to be 70.7% for this algorithm which is acceptable in terms of whether the model is able to even distinguish between the two class labels (job seeker and non-job seeker) or not. The model presented in this project is intended to apply only to people who have enrolled in some training courses. It is not applicable to any student or any industry professional who is not pursuing any kind of vocational training.

REFERENCES

1. Turban, D. B., & Keon, T. L. (1993). *Organizational attractiveness: An interactionist perspective.* Journal of Applied Psychology, 78*(2), 184–193.*
2. E. Barber M. V. Roehling (1993). *Job Postings and The Decision to Interview: A Verbal Protocol Analysis,* Journal of Applied Psychology 78 845–856
3. Khan, Naveed & Awang, Marinah & Ghouri, Arsalan. (2013). *Impact Of E-Recruitment and Job-Seekers Perception on Intention to Pursue the Jobs*, Management & Marketing. 11. 47-57.
4. Wanberg, C. R., Kanfer, R., & Rotundo, M. (1999). *Unemployed individuals: Motives, job-search competencies, and job-search constraints as predictors of job seeking and reemployment.* Journal of Applied Psychology, 84(6), 897–910.
5. Hooft, Edwin & Jong, Mireille. (2009). *Predicting Job Seeking for Temporary Employment Using the Theory of Planned Behaviour: The Moderating Role of Individualism and Collectivism.* Journal of Occupational and Organizational Psychology. 82. 295-316.
6. Gatewood, R., Gowan, M., & Lautenschlager, G. (1993). *Corporate Image, Recruitment Image, and Initial Job Choice Decisions.* The Academy of Management Journal, 36(2)
7. Catanzaro, D., Moore, H., & Marshall, T. (2010). *The Impact of Organizational Culture on Attraction and Recruitment of Job Applicants.* Journal of Business and Psychology, 25(4), 649-662
8. Hooft, Edwin & Kammeyer-Mueller, John & Wanberg, Connie & Kanfer, Ruth & Basbug, Gokce. (2020). *Job Search and Employment Success: A Quantitative Review and Future Research Agenda.*

9. Bolliger, Doris U. (2004), *Key Factors for Determining Student Satisfaction in Online Courses.* International Journal on E-Learning, 3*(1), 61-67.*
10. Nonis, Sarath & Fenner, Grant. (2021). *An Exploratory Study of Student Motivations for Taking Online* Courses and Learning Outcomes.
11. Lim, *R. H., Lent, R. W., & Penn, L. T. (2016). Prediction of job search intentions and behaviors: Testing the social cognitive model of career self-management.* Journal of Counseling Psychology, 63*(5), 594–603.*
12. Tan, P.N., Steinbach, M., Karpatne, A. and Kumar, V, (2019), *Introduction to Data Mining*, New York, NY: Pearson Education, Inc.