Campus Placements Prediction & Analysis using Machine Learning

Manav Aditya Sinha
Tannu Rana
Nisha
Muskaan
Ishita Yadav
Rudrayani
(SOET, Department of Artificial Intelligence & Machine Learning)
KR Mangalam University

Abstract-

Campus placement is an activity of participating, identifying and hiring young talent for internships and entry level positions.

Reputation and yearly admissions of the institute invariably depend upon the placements provided by the institute to the students. Therefore, most of the institutions, assiduously, try to boost their placement department in order to improve their organization on a full scale. Any assistance during this specific space can have a good impact on the institute's capability to position it's students.

In this study the target is to analyze student's placement data of given data and it to determine the probability of campus placement of present students. For this we have experimented with six different machine learning algorithms i.e. Logistic Regression, LDA, Random Forest, K Nearest Neighbor, SVS, Naive Byes.

I. INTRODUCTION

NOWDAYS the number of educational institutes is growing day by day. The aim of each higher educational institute is to help their students to get a well-paid job through their placement cell. One of the biggest challenges that higher learning institutes face these days is to uplift the placement performance of scholars.

The goal of this system is to predict whether the student will get a placement or not based on various parameters such as secondary percentage, secondary branch, high school percentage, high school branch, high school subject, degree percent, degree type, work experience, employment test percent, specialization & MBA percentage.

This research focuses on various algorithms of machine learning such as Logistic Regression, LDA, Random Forest, K Nearest Neighbor, SVS and Naive Byes in order to produce economical and correct results for campus placement prediction. This system follows a supervised machine learning approach as it uses class labelled data for training the classification algorithm.

II. METHODOLOGY

The steps involved in this system are as follows,

A. Data Acquisition:

The campus placement dataset is collected from Kaggle website. Here is the link for the dataset: https://www.kaggle.com/c/ml-with-python-course-project/data.csv

The dataset consists of various attributes such as Serial Number, Gender, SSC percentage, SSC Board - Central/ Others, HSC percentage, HSC Board, HSC Specialization, Degree Percentage, UG Degree Stream, Work Experience, E -test Percentage, Degree Specialization, Degree Percentage, Placement Status & Salary. The size of dataset is 19.71 KB & it has total 215 records.

1) Handling missing values:

In our dataset missing values are present only in the salary column as these values correspond to the students who didn't get placed in any placement drive. So it is assumed that the missing values in Salary Column are Zero & replaced them by zero using fillna(0,inplace=True) function in Python.

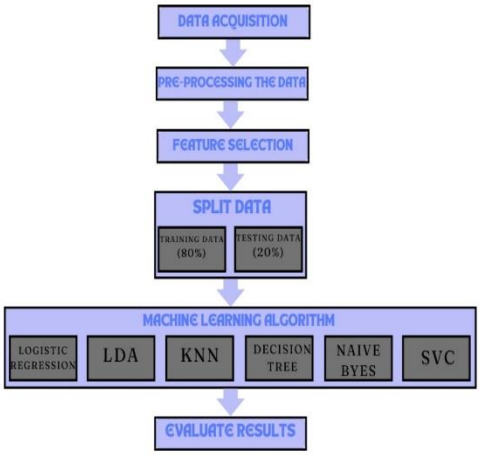
2) Handling categorical data:

Since we cannot deal with categorical values directly, mapping is done for attributes having categorical values.

Gender attribute has values M (Male) & Female (M). Here, M is replaced by 0 & F is replaced by 1. SSC & HSC Board attributes has values 'Central' & 'Other.' Here, Central is replaced by 1 & Other is replaced by 0. Work Experience attribute has values 'Yes' & 'No'. Here, 'Yes' is replaced by 1 and 'No' is replaced by 0. Degree specialization attribute has values 'Marketing & Finance' & 'Marketing & HR'. Here, 'Marketing & Finance' is replaced by 1 and 'Marketing & HR' is replaced by 0. Status attribute has values 'Placed' and 'Not Placed'. Here, 'Placed' is replaced by 1 and 'Not Placed' is replaced by 0. This is achieved through map function in Python.

For e.g.,

- df['gender']=df['gender'].map({'M':0,'F':1})
- df['ssc_b']=df['ssc_b'].map({'Central':1,'Others':0})
- df['workex']=df['workex'].map({'Yes':1,'No':0})



3) Feature selection:

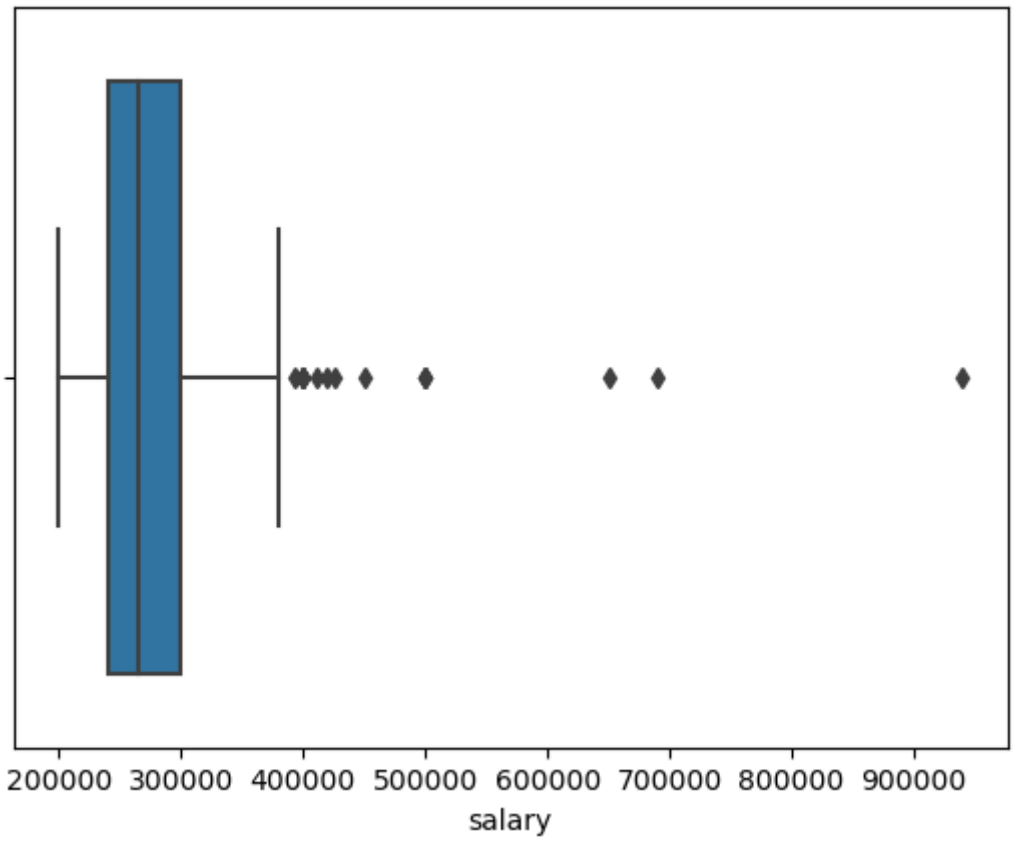
Here, various features are visualized to understand their correlation with the target feature.

Boxplot:

A graph that gives a visual indication of how a data set's 25th percentile, 50th percentile, 75th percentile, minimum, maximum and outlier values are spread out and compare to each other.

This topic describes the Box Plot component provided by Machine Learning Designer. A box plot chart shows the distribution of a set of data. It shows the distribution features of raw data. It can also be used to compare the distribution features between multiple sets of data.

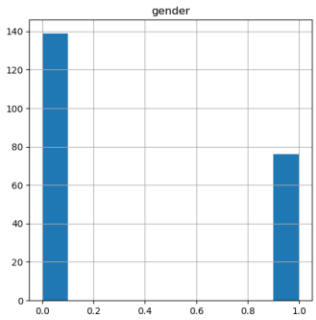
A box and whisker plot or diagram (otherwise known as a boxplot), is a graph summarising a set of data. The shape of the boxplot shows how the data is distributed and it also shows any outliers. It is a useful way to compare different sets of data as you can draw more than one boxplot per graph.

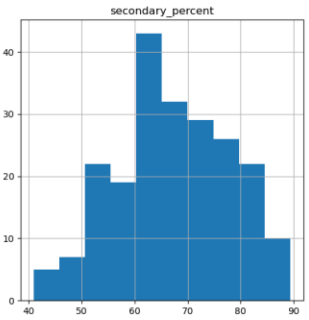


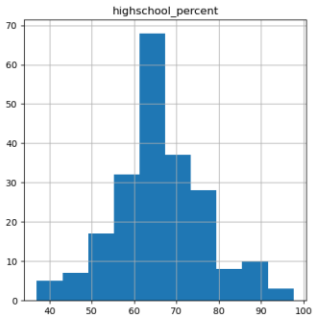
Histogram:

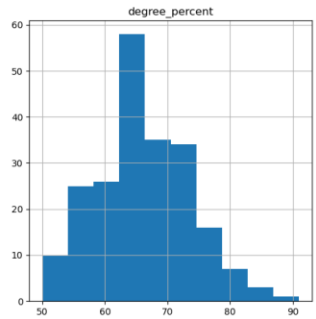
A histogram is used to check the shape of the data distribution. Used to check whether the process changes from one period to another. Used to determine whether the output is different when it involves two or more processes. Used to analyse whether the given process meets the customer requirements.

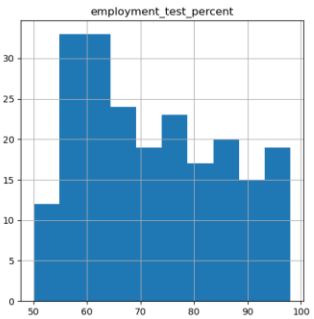
The histogram is a popular graphing tool. It is used to summarize discrete or continuous data that are measured on an interval scale. It is often used to illustrate the major features of the distribution of the data in a convenient form.

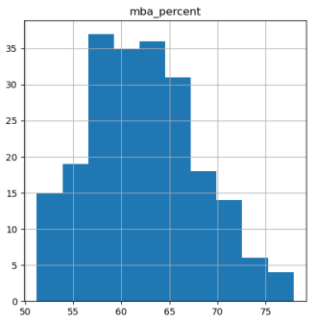


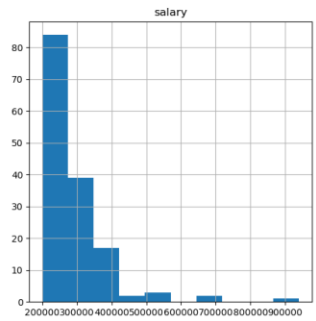








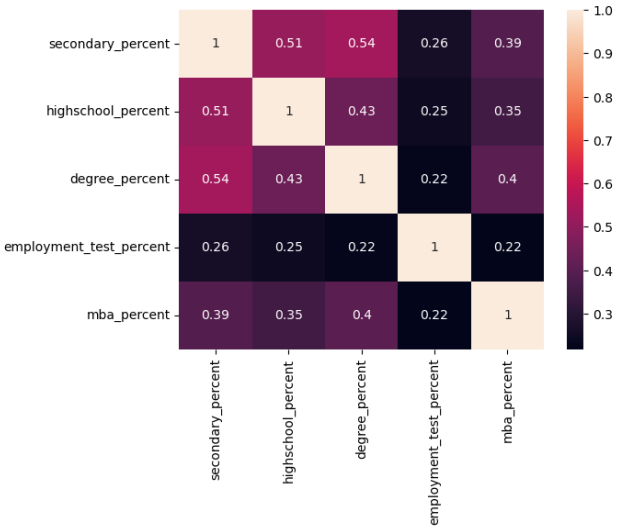




Heatmap:

Heat maps in Machine Learning are commonly used to compare the performance of various models or algorithms on a given task or dataset. For instance, you can use them to assess the accuracy scores of different models on different subsets or folds of data, which will help you determine the best model for your problem.

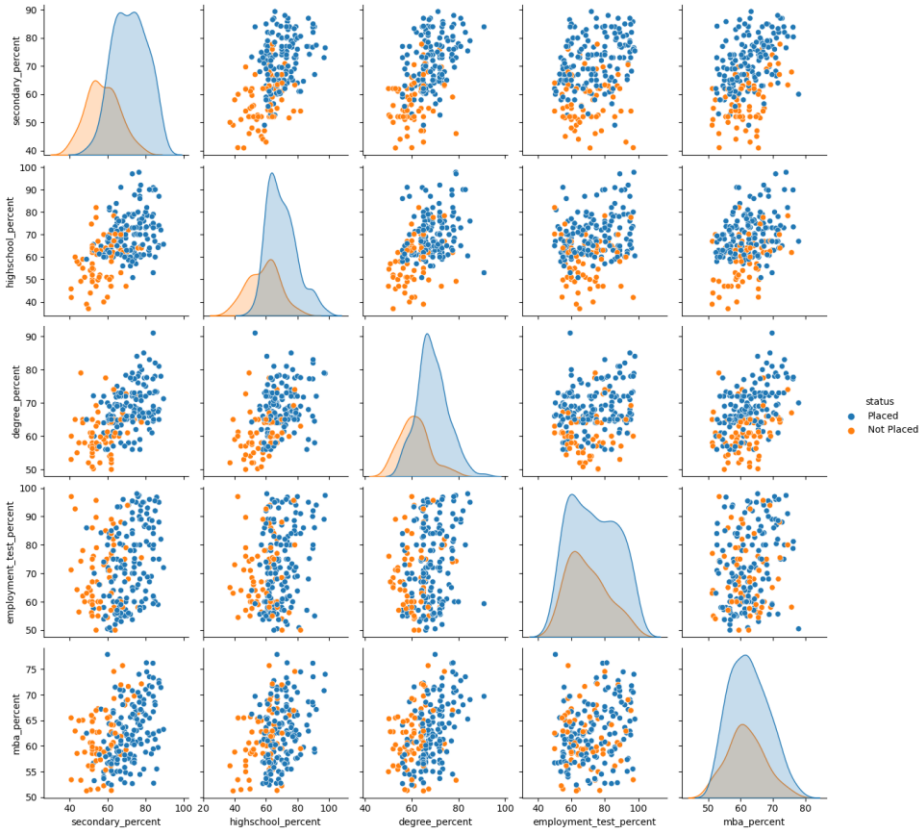
A heatmap (aka heat map) depicts values for a main variable of interest across two axis variables as a grid of colored squares. The axis variables are divided into ranges like a bar chart or histogram, and each cell's color indicates the value of the main variable in the corresponding cell range.



Pair plot:

A pair plot, also known as a scatterplot matrix, is a matrix of graphs that enables the visualization of the relationship between each pair of variables in a dataset. It combines both histogram and scatter plots, providing a unique overview of the dataset's distributions and correlations.

Pair plot is used to describe pairwise relationships in a dataset. Pair plot is used to visualize the univariate distribution of all variables in a dataset along with all of their pairwise relationships. For n variables, it produces n*n grid. The diagonal plots are histograms and all the other plots are scatter plots.



4) Split data:

Here, data is divided into two parts i.e. training data & testing data. Where 80 % data is taken for training our machine learning algorithm and remaining 20 % data is used for testing whether our trained machine learning model is working correctly or not.

5) Machine Learning Algorithm:

a) Logistic Regression:

Logistic regression is a statistical method used to determine the outcome of a dependent variable (y) based on the values of independent variable (x).

In our problem dependent variable is placement status and independent variables are the features selected by us in the previous step. This algorithm is mostly used for the problems of binary classification.

b) Decision Tree:

A decision tree is a graph like a tree where nodes represent the position where we select the feature and ask a question, edges represent the answers of the question; and the leaves represent the final output or label of the class.

c) KNN:
K-NN stores all the training data into different classes based on the class labels and classifies new data by checking its similarity with data in the available classes.

d) Linear Discriminant Analysis:
It is an approach used in supervised machine learning to solve multi-class classification problems. LDA separates multiple classes with multiple features through data dimensionality reduction. This technique is important in data science as it helps optimize machine learning problems.

e) Naïve Byes:
In simple terms, a Naïve Byes classifier assumes that the presence of a particular feature in class is unrelated to the presence of any other feature. The Naïve Byes classifier is a popular supervised machine learning algorithm used for classification tasks such as text classification.

f) Support Vector Classifier (SVC):
SVC is a specific implementation of the Support Vector Machine algorithm that is designed specifically for classification tasks. In other words, SVC is an SVM used for classification. It seeks to find the hyperplane that best separates the data points into different classes.

6) Evaluate results:

Accuracy is calculated by following formula:
Accuracy = (TP + TN) / (TP + FP + TN + FN)
Where:
TP: True Positive (the number of cases correctly identified as placed)
TN: True Negative (the number of cases correctly identified as unplaced)
FP: False Positive (the number of cases incorrectly identified as placed)
FN: False Negative (the number of cases incorrectly identified as unplaced)

OBSERVATION: LDA has the highest accuracy of 93%

III. CONCLUSION

The problem of campus placement prediction can be solved with the help of different machine learning algorithms such as Logistic regression, Decision Tree, KNN & Random Forest.
Here, the Logistic Regression algorithm gave the highest accuracy of 95. 34 % for campus placements prediction.
The selected features i.e. Gender, SSC percentage, SSC Board - Central/ Others, HSC percentage, HSC Board, HSC Specialization, Degree Percentage, UG Degree Stream, Work Experience, E -test Percentage, Degree Specialization & Degree Percentage lead to higher classification accuracy.

IV. FUTURE SCOPE

Accuracy may further increase by application of more advanced techniques such as deep learning & experimenting with different activation functions of neural networks such as linear, sigmoid, tan h & ReLU.
We can also experiment with different cross validation techniques such as 3-Fold, 5 -Fold, 10 -Fold, 15-Fold cross validation in order to analyse the change in accuracy.