|  |  |
| --- | --- |
| Name Of The Student | Deepa D. |
| Internship Project Topic | RIO-125-HR Salary Dashboard- Train the Dataset and Predict Salary |
| Name of the Organization | TCS iON |
| Name of the Industry Mentor | Debashis Roy |
| Name of the Institute | ICT ACADEMY OF KERALA |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Date | End Date | | Total Effort (hrs.) | | Project Environment | Tools used |
| 13-02-2022 | 31-03-2022 | | 125 hours | | Python Environment | Google Colabatory |
| Milestone # | 3 | Milestone: | | Create data set, Clean and sanitize dataset, Preprocessing data set, Test and train the dataset, Build the classifier models and fit the data in the model, Dash board creation. | | |

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| **Acknowledgement** |  |
| 1. **Objective** |  |
| 1. **Introduction** |  |
| 1. **Internship Activities**    1. Creating the dataset    2. Analyzing the dataset    3. Cleaning and sanitizing    4. Splitting the data set    5. Scaling    6. Modeling    7. Fine tuning 2. **Result & Conclusion** 3. **Enhancement scope** |  |
| 1. **Link to the project code** |  |

**ACKNOWLEDGMENTS**

I am conveying my sincere gratitude towards my industry mentor, Debashis Roy, and academic mentor, Aswathy P., for helping me throughout this project till now and providing me this wonderful platform to complete this project. I am also thankful for answering my queries at every phase of the project. I also want to thank all my friends who helped me with valuable suggestions during this project.

**1.OBJECTIVE**

The Objective of this project is to build salary prediction dashboard for Human Resource Management. In order to select a right candidate for a job position, along with assessing the credentials and qualifications of the candidates, HR should consider the previous salary offered to similar candidates based on their experience, age, qualifications etc. Since the salary is affected by many factors, using algorithms to predict salary is a good idea and machine learning models are very useful.

**2.INTRODUCTION**

Machine Learning is emerging as an advanced technology that is supporting organizations within the domains of organizational aspects, people management, and business strategies. There has been an evident escalation in Human Resources in recent times, as the market is experiencing a boom in quality resources and skills which therefore is a competitive incentive for enterprises to invest in Proper recruiting has become more crucial than ever and is an essential element in the enterprise’s corporate plans due to its consequences on the businesses competitiveness and efficiency. This report put an emphasis on predicting the employee’s new salary based on several factors such as age, work class, education, experience, previous occupation, past income, hours-per-week. Based on the above dataset parameters, salary can be predicted accurately. The human resource department is facing a tough job nowadays to process a huge number of applicants and at the same time, they need to select the right candidates for the right position. Candidates usually considering salary as one of the defining factors while accepting a job offer. Hence offering the right and comparable salary in the market to the candidates are very important.

Dataset selection/ importing always depends on the type of algorithm used, whether supervised or unsupervised or semi-supervised and also the number of valid records and attributes must be considered important selection criteria. There are different datasets available over the internet. However, by considering the prerequisites, this work analyses a well-respected dataset having 32562 salary data with 14 columns. It represents a wide range of job profiles, experience, and salary hence it is a good dataset for our analysis. It is in CSV format. The attributes referred to in the dataset are logical and competent to have a good idea in predicting the new salary efficiently and accurately.

The target salary in the dataset I have selected contains only two classes (<=50K and >50K ). Hence, the model needs to predict the salary to be in one of these two classes. So, our model converges to a binary classification model. There are several methods to make a binary classification, which include Gradient Boosting, logistic regression, random forest, etc. So I have trained and tested my data using Logistic Regression, KNN, Decision tree, Gradient boost and Random Forest models, compared them to select the best model.

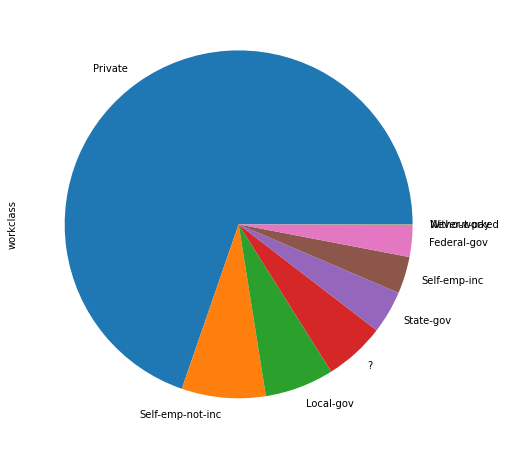
**3.INTERNSHIP ACTIVITIES**

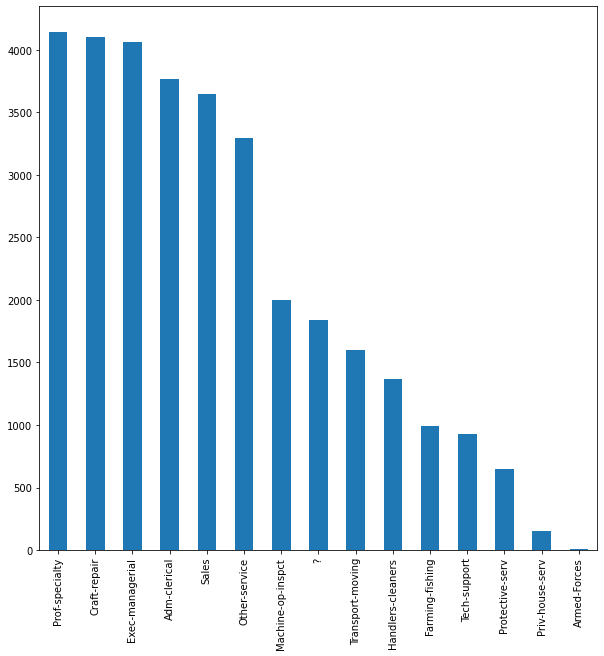
**3.1 Creating the dataset**

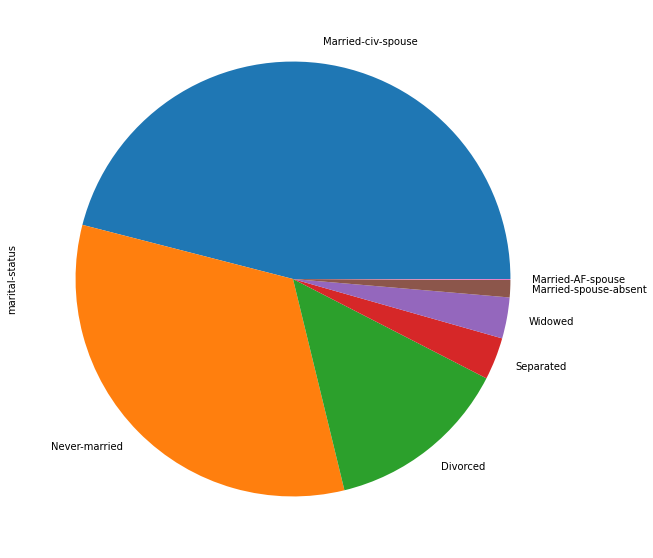
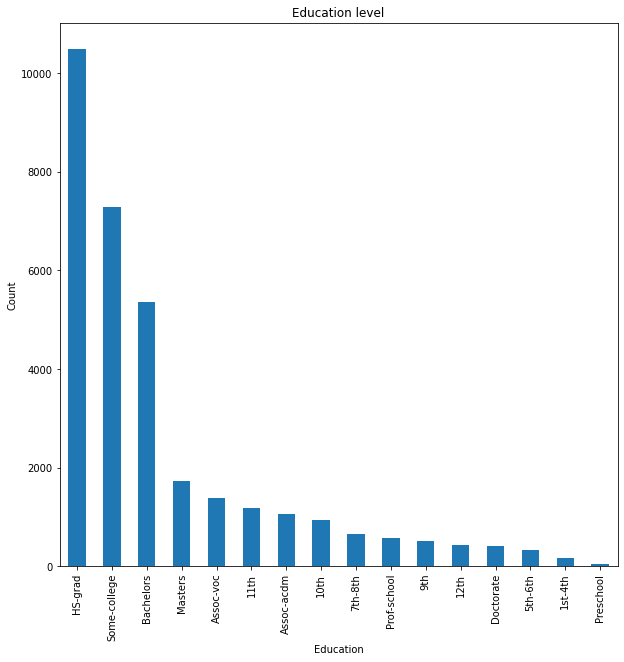
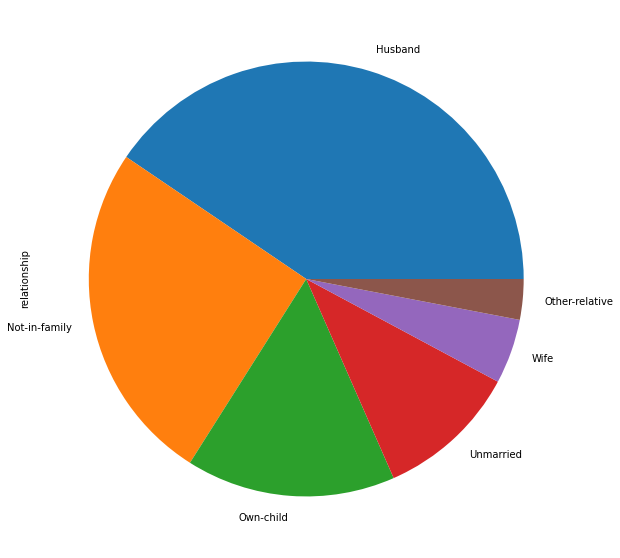
Selected a dataset with 32562 rows and 14 columns for this project. The columns include Age, Work class, Education, Education number, Marital-status, Occupation, Relationship, Race, Sex, Capital-gain, Capital-loss, hours per week, native country & salary.

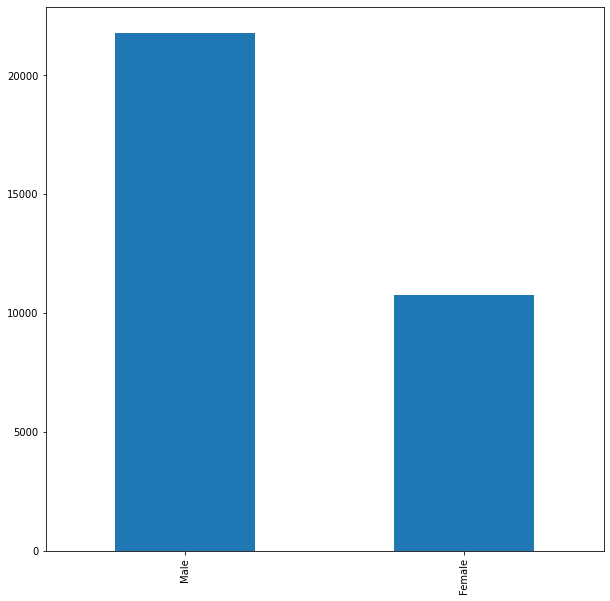
**3.2 Analyzing the dataset**

When analyzing the dataset, it is clear that Capital-loss and Capital- gain have no role in predicting the salary. When I checked what are these attributes for an employee, I came to know that these columns incur only when that particular person has a salary. So, we can drop it. Similarly, we can drop the column Education number, since it is a numerical version of the column Education. Exploratory data analysis of different features in the dataset are given below.









* 1. **Cleaning and sanitize the dataset**

In the data set, no null values are present, but ‘?’ present in some columns and are replace with respective mode of each column. Then checked for the outliers before processing, and in our data, outliers are found in some of the columns like age and hours-per-week and are removed. The categorical data has been converted to a numerical form in this step using label encoding.

* 1. **Splitting the data**

The data has been split into two before scaling the data.

* 1. **Scaling of data**

Many machine learning algorithms perform better when numerical input variables are scaled to a standard range The two most popular techniques for scaling numerical data prior to modeling are normalization and standardization.

**Normalization** scales each input variable separately to the range 0-1, which is the range for floating-point values where we have the most precision.

**Standardization** scales each input variable separately by subtracting the mean (called centering) and dividing by the standard deviation to shift the distribution to have a mean of zero and a standard deviation of one.

**Standardization is used here in our dataset.**

* 1. **Modeling**

In machine learning, classification refers to a predictive modeling problem where a class label is predicted for a given example of input data. A classification model tries to draw some conclusions from the input values given for training. It will predict the class labels/categories for the new data.

1. Logistic regression classifier

Logistic regression is a machine learning algorithm for classification. In this algorithm, the probabilities describing the possible outcomes of a single trial are modeled using a logistic function. Logistic regression is designed for this purpose (classification) and is most useful for understanding the influence of several independent variables on a single outcome variable.

It works only when the predicted variable is binary, assumes all predictors are independent of each other, and assumes data is free of missing values.

1. **K-Nearest Neighbours**

Neighbors-based classification is a type of lazy learning as it does not attempt to construct a general internal model but simply stores instances of the training data. Classification is computed from a simple majority vote of the k nearest neighbors of each point. This algorithm is simple to implement, robust to noisy training data, and effective if training data is large. Need to determine the value of K and the computation cost is high as it needs to compute the distance of each instance to all the training samples

1. Decision Tree classifier

Given a data of attributes together with its classes, a decision tree produces a sequence of rules that can be used to classify the data. [Decision Tree](https://analyticsindiamag.com/hands-on-tutorial-how-to-use-decision-tree-regression-to-solve-machinehacks-new-data-science-hackathon/) is simple to understand and visualize, requires little data preparation, and can handle both numerical and categorical data.

A decision tree can create complex trees that do not generalize well, and decision trees can be unstable because small variations in the data might result in a completely different tree being generated.

1. Random forest classifier

[Random forest](https://analyticsindiamag.com/step-by-step-guide-to-reviews-classification-using-svc-naive-bayes-random-forest/) classifier is a meta-estimator that fits a number of decision trees on various sub-samples of datasets and uses an average to improve the predictive accuracy of the model and controls over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement. Reduction in over-fitting and random forest classifiers is more accurate than decision trees in most cases.

The disadvantages of a Random Forest classifier are real-time prediction, difficulty to implement, and complex algorithm.

1. Gradient Boost classifier

Gradient boosting classifiers are a group of machine learning algorithms that combine many weak learning models together to create a strong predictive model. Decision trees are usually used when doing gradient boosting. The idea behind "gradient boosting" is to take a weak hypothesis or weak learning algorithm and make a series of tweaks to it that will improve the strength of the hypothesis/learner. This type of Hypothesis Boosting is based on the idea of [Probability Approximately Correct Learning](https://en.wikipedia.org/wiki/Probably_approximately_correct_learning) (PAC).This PAC learning method investigates machine learning problems to interpret how complex they are, and a similar method is applied to Hypothesis Boosting. In hypothesis boosting, you look at all the observations that the machine learning algorithm is trained on, and you leave only the observations that the machine learning method successfully classified behind, stripping out the other observations. A new weak learner is created and tested on the set of data that was poorly classified, and then just the examples that were successfully classified are kept. This idea was realized in the [Adaptive Boosting](https://towardsdatascience.com/boosting-algorithm-adaboost-b6737a9ee60c) (AdaBoost) algorithm. For AdaBoost, many weak learners are created by initializing many decision tree algorithms that only have a single split, such as the "stump" in the image below.

Five different models are tried and compared the results.

|  |  |
| --- | --- |
| Model | Accuracy |
| Logistic Regression | 77.32 |
| KNN | 82.84 |
| Decision Tree | 79.75 |
| Random Forest | 83.53 |
| Gradient Boost | 85 |

When compared the results, Gradient Boost modeling have better accuracy than other classifiers. So, we just used hyperparameter tuning to check whether the performance is improved or not.

* 1. **Fine Tuning**

Fine tuning machine learning predictive models is crucial step to improve accuracy of the forecasted results. Most performance variation can be attributed to just a few hyperparameters; the tunability of an algorithm, hyperparameter, or interacting hyperparameters is a measure of how much performance can be gained by tuning it. In machine learning, a hyperparameter is a parameter whose value is used to control the learning process. The values of other parameters are derived via training.

Here I have done fine tuning of Gradient Boost modeling and got an accuracy above 86.36%. Hence the accuracy is improved and decided to use Random Forest classifier as model for HR salary prediction.

1. **RESULTS & CONCLUSION**

There are 14 features in our salary prediction dataset and found that only 11 features contributed to the prediction. We used different models and highest accuracy was obtained for Gradient Boost Classifier. So Gradient Boost classifier is fine-tuned, and performance is improved. So Gradient Boost classifier is selected as the best model for this dataset for predicting the HR salary prediction. A website is hosted using these features to predict salary.

1. **ENHANCEMENT SCOPE**

This industry project has a wide scope. Using the resume or CV of an individual, one can actually predict the salary. Some of the Natural language processing techniques will help in developing this application

1. **Link to the project code**

**Colab link** [**https://colab.research.google.com/drive/1vIP6kaQIX0EwWLRa18TZ82DCDAeOEHCj?usp=sharing**](https://colab.research.google.com/drive/1vIP6kaQIX0EwWLRa18TZ82DCDAeOEHCj?usp=sharing)

**Github link** [**https://github.com/DeepaDolly/HR-Salary-prediction-with-python-flask**](https://github.com/DeepaDolly/HR-Salary-prediction-with-python-flask)